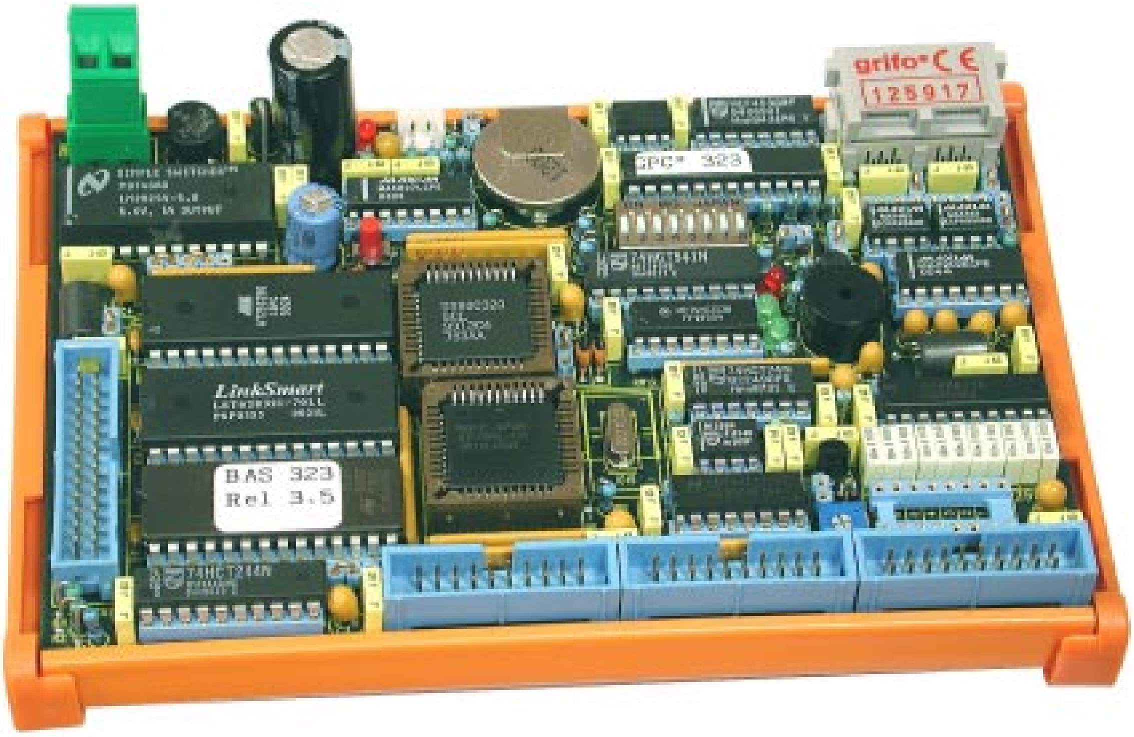
GPC® 323

General Purpose Controller 80c32, 80c320, 89c51Rx2

MANUALE TECNICO



grifo®
ITALIAN TECHNOLOGY
Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY
E-mail: grifo@grifo.it
http://www.grifo.it http://www.grifo.com
Tel. +39 051 892.052 (r.a.) FAX: +39 051 893.661

GPC® 323 Edizione 3.30 *Rel. 23 Gennaio 2002*

GPC® 323

General Purpose Controller 80c32, 80c320, 89c51Rx2

MANUALE TECNICO

Modulo intelligente della serie **ABACO®** BLOCK, nel formato 100x149; contenitore opzionale per guide ad Ω tipo DIN 46277-1 e DIN 46277- 3; CPU 80C32, 80C320 o 89c51Rx2 a 22 MHz con indirizzamento massimo esterno di **96K Bytes** con CPU 80c32 o 80c320 e fino a 128K Bytes con **P89c52RD2**; zoccoli per **32K EPROM**, **32K SRAM**, **32 EEPROM, SRAM, FLASH o EPROM**; circuiteria di **Back Up** per 32K SRAM, tramite **batteria al LITIO** esterna; **EEPROM seriale** opzionale da 512 a **2048 Bytes**; **RTC** opzionale con batteria al LITIO e 256 Bytes di SRAM; 11 linee di **A/D Converter** da **12 Bits**, +2,490V o +5,000 V di fondo scala oppure 0÷20 mA, tempo di **conversione 10 μs**; 24 linee di **I/O TTL**;1 **Dip Switch** con 5 vie leggibili da software; **3 Timer Counter** da 16 bits con registri di Capture e comparazione; circuiteria di **Watchdog** con LED di segnalazione; **2 linee serial in RS232**, di cui una settabile in **RS 422**, **RS 485** o **Current Loop**; connettore di espansione **ABACO®** I/O BUS da 26 vie; 2 connettori standard di I/O **ABACO®** da 20 vie; 1 connettore standard di A/D **ABACO®** da 20 vie; funzionamento in IDLE MODE o POWER DOWN MODE; alimentatore switching incorporato **8÷24 Vac** (12÷34 Vdc) opzionale o alimentazione esterna da **+5 Vdc**; **Protezione della logica** di bordo dai transienti tramite **TransZorb™**; 2 **LEDs di stato**; 1 **buzzer** attivo; vasta disponiblità di linguaggi di programmazione quali **BASIC 323**, **BASCOM 8051**, **DDS C**, **HI TECH C**, ecc.

grifo®
ITALIAN TECHNOLOGY

Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY
E-mail: grifo@grifo.it
http://www.grifo.it http://www.grifo.com
Tel. +39 051 892.052 (r.a.) FAX: +39 051 893.661

GPC® 323 Edizione 3.30 *Rel. 23 Gennaio 2002*

IMPORTANTE

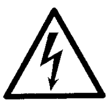
Tutte le informazioni contenute sul presente manuale sono state accuratamente verificate, ciononostante **grifo®** non si assume nessuna responsabilità per danni, diretti o indiretti, a cose e/o persone derivanti da errori, omissioni o dall'uso del presente manuale, del software o dell' hardware ad esso associato.
grifo® altresi si riserva il diritto di modificare il contenuto e la veste di questo manuale senza alcun preavviso, con l' intento di offrire un prodotto sempre migliore, senza che questo rappresenti un obbligo per **grifo®**.
Per le informazioni specifiche dei componenti utilizzati sui nostri prodotti, l'utente deve fare riferimento agli specifici Data Book delle case costruttrici o delle seconde sorgenti.

LEGENDA SIMBOLI

Nel presente manuale possono comparire i seguenti simboli:



Attenzione: Pericolo generico



Attenzione: Pericolo di alta tensione

Marchi Registrati

INDICE GENERALE

INDICE DELLE FIGURE

INTRODUZIONE

L'uso di questi dispositivi é rivolto - IN VIA ESCLUSIVA - a personale specializzato.

Scopo di questo manuale é la trasmissione delle informazioni necessarie all'uso competente e sicuro dei prodotti. Esse sono il frutto di un'elaborazione continua e sistematica di dati e prove tecniche registrate e validate dal Costruttore, in attuazione alle procedure interne di sicurezza e qualità dell'informazione.

I dati di seguito riportati sono destinati - IN VIA ESCLUSIVA - ad un utenza specializzata, in grado di interagire con i prodotti in condizioni di sicurezza per le persone, per la macchina e per l'ambiente, interpretando un'elementare diagnostica dei guasti e delle condizioni di funzionamento anomale e compiendo semplici operazioni di verifica funzionale, nel pieno rispetto delle norme di sicurezza e salute vigenti.

Le informazioni riguardanti installazione, montaggio, smontaggio, manutenzione, aggiustaggio, riparazione ed installazione di eventuali accessori, dispositivi ed attrezzature, sono destinate - e quindi eseguibili - sempre ed in via esclusiva da personale specializzato avvertito ed istruito, o direttamente dall'ASSISTENZA TECNICA AUTORIZZATA, nel pieno rispetto delle raccomandazioni trasmesse dal costruttore e delle norme di sicurezza e salute vigenti.

I dispositivi non possono essere utilizzati all'aperto. Si deve sempre provvedere ad inserire i moduli all'interno di un contenitore a norme di sicurezza che rispetti le vigenti normative. La protezione di questo contenitore non si deve limitare ai soli agenti atmosferici, bensì anche a quelli meccanici, elettrici, magnetici, ecc.

Per un corretto rapporto coi prodotti, é necessario garantire leggibilità e conservazione del manuale, anche per futuri riferimenti. In caso di deterioramento o più semplicemente per ragioni di approfondimento tecnico ed operativo, consultare direttamente l'Assistenza Tecnica autorizzata.

Al fine di non incontrare problemi nell'uso di tali dispositivi, é conveniente che l'utente - PRIMA DI COMINCIARE AD OPERARE - legga con attenzione tutte le informazioni contenute in questo manuale. In una seconda fase, per rintracciare più facilmente le informazioni necessarie, si può fare riferimento all'indice generale e all'indice analitico, posti rispettivamente all'inizio ed alla fine del manuale.

VERSIONE SCHEDA

Il presente manuale è riferito alla scheda **GPC® 323** versione **250601** e successive. La validità delle informazioni riportate è quindi subordinata al numero di versione della scheda in uso e l'utente deve quindi sempre verificare la giusta corrispondenza tra le due indicazioni. Sulla scheda il numero di versione è riportato in più punti sia a livello di serigrafia che di stampato (ad esempio nell'angolo in basso a destra sopra C69 nel lato componenti o sul lato destro sotto IC8 nel lato saldature).

CARATTERISTICHE GENERALI

La scheda **GPC® 323** é un potente modulo di controllo, della fascia **Low-Cost**, in grado di funzionare autonomamente come periferica intelligente e/o remotata in una più vasta rete di telecontrollo e/o di acquisizione.
La **GPC® 323** é fornita di un supporto in plastica provvisto degli attacchi per le guide ad **Omega** tipo **DIN 46277-1** e **DIN 46277-3**. In questo modo non é necessario l'uso di un rack, perché la scheda può essere montata, in modo più economico, direttamente nel quadro elettrico.
L'aspetto più interessante di questa scheda é data dal fatto che può essere equipaggiata da una vasta schiera di µP. E' infatti possibile averla con il normale **80C32** o con il velocissimo **DALLAS 80C320**. Montando un microcontrollore del tipo **PHILIPS P89c51Rx2** è inoltre possibile programmarne la FLASH interna con la tecnica **ISP**, evitando così di dovere ogni volta rimuovere il chip dalla scheda. I microcontrollori tipo P89c51Rx2 esistono in quattro differenti taglie di FLASH interna, da 16K a 64K Byte. Le caratteristiche della scheda rimangono fondamentalmente le stesse pur variando notevolmente in prestazioni a seconda del tipo di µP montato.
Tramite linguaggi ad alto livello è possiblie un uso immediato di interfacce operatore; sono disponibili delle schede della serie **KDx x24** oppure, per chi ha bisogno di un oggetto finito, esistono le **QTP xxP**. Questi pannelli operatore, offerti nella versione a giorno, hanno la stessa estetica della **QTP xx** ma, non disponendo di intelligenza locale, sono comandati direttamente dalla **GPC® 323**, consentendo così una notevole riduzione dei costi.
Dispone inoltre di varie soluzioni di alimentazione in modo da poter essere facilmente installata in tutti i sistemi di controllo senza dover provvedere ad apposite e costose sezioni alimentatrici esterne.
La **GPC® 323** é dotata di una serie di connettori normalizzati, standard **Abaco®**, che le consentono di utilizzare immediatamente la numerosa serie di moduli **BLOCK** di I/O oppure permettono il collegamento, in modo molto semplice ed economico, delle interfacce da campo costruite direttamente dall'utente o da terze parti.
La disponibilità del connettore **ABACO® I/O BUS** consente inoltre di poter pilotare direttamente le schede tipo **ZBR 324, ZBT 324, ABB 03, ABB 05** e quindi le numerose schede periferiche del carteggio **grifo®**, espandendo rapidamente le caratteristiche di base della scheda.
Attualmente sono disponibili diversi Tools di sviluppo software che consentono di poter usare la scheda come sistema di sviluppo di se stessa, sia in Assembler che con linguaggi evoluti. Una particolare menzione và ai tools di sviluppo quali i vari **Compilatori C**, il **FORTH**, i comodi compilatori basic **BASCOM 8051** e **BASIC 323**.

- Modulo intelligente della serie **ABACO®** BLOCK, nel formato 100x149
- Contenitore opzionale per guide ad Ω tipo DIN 46277-1 e DIN 46277- 3
- CPU 80C32, 80C320 o 89c51Rx2 a 22 MHz
- Indirizzamento massimo esterno di **96K Bytes** con CPU 80c32 o 80c320 e fino a 128K Bytes con **P89c52RD4**
- Zoccoli per **32K EPROM**, **32K SRAM**, **32 EEPROM, SRAM, FLASH o EPROM**
- Circuiteria di **Back Up** per 32K SRAM, tramite **batteria al LITIO** esterna
- **EEPROM seriale** opzionale da 512 a **2048 Bytes**
- **RTC** opzionale con batteria al LITIO e 256 Bytes di SRAM
- 11 linee di **A/D Converter** da **12 Bits**, +2,490V o +5,000 V di fondo scala oppure 0÷20 mA, tempo di **conversione 10 µs**
- 24 linee di **I/O TTL**
- 1 **Dip Switch** con 5 vie leggibili da software
- **3 Timer Counter** da 16 bits con registri di Capture e comparazione
- Circuiteria di **Watchdog** con LED di segnalazione

- **2 linee seriali in RS 232**, di cui una settabile in **RS 422**, **RS 485** o **Current Loop**
- Connettore di espansione **ABACO®** I/O BUS da 26 vie
- 2 connettori standard di I/O **ABACO®** da 20 vie
- 1 connettore standard di A/D **ABACO®** da 20 vie
- Funzionamento in IDLE MODE o POWER DOWN MODE
- Alimentatore switching incorporato **8÷24 Vac** (12÷34 Vdc) opzionale o alimentazione esterna da **+5 Vdc**
- **Protezione della logica** di bordo dai transienti tramite **TransZorb™**
- 2 **LEDs di stato**
- 1 **buzzer** attivo
- Vasta disponiblità di linguaggi di programmazione quali **BASIC 323**, **BASCOM 8051**, **DDS C**, **HI TECH C**, etc..

Viene di seguito riportata una descrizione dei blocchi funzionali della scheda, con indicate le operazioni effettuate da ciascuno di essi. Per una più facile individuazione di tali blocchi e per una verifica delle loro connessioni, fare riferimento alla figura 1.

CONTATTO DI RESET

Sulla **GPC® 323** è presente un contatto di reset (P1) che ha la funzione di resettare e quindi far ripartire la scheda da una condizione di azzeramento generale. La funzione principale di questo contatto è quella di uscire da condizioni di loop infinito, soprattutto durante la fase di debug o di garantire uno stato certo di partenza. Per una facile individuazione di tale contatto di reset a bordo scheda, si faccia riferimento alla figura 9.

COMUNICAZIONE SERIALE

La **GPC® 323** dispone sempre di una linea seriale hardware (definita seriale A) ed una seconda linea seriale (definita seriale B) gestita come segue:

µP 80C32: seriale software gestita tramite due linee di I/O del microprocessore;
µP 80C320: seriale hardware gestita tramite registri interni del microprocessore;

Il protocollo fisico (baud rate, stop bit, bit x chr, ecc.) delle linee seriali hardware é completamente settabile via software tramite la programmazione dei registri interni al microcontrollore di cui la scheda è provvista, quindi per ulteriori informazioni si faccia riferimento alla documentazione tecnica della casa costruttrice o alle appendici di questo manuale. Per la linea seriale software il protocollo fisico é invece direttamente definito da apposite procedure che agiscono direttamente sulla linea di ricezione e trasmissione. Alcuni pacchetti software (come il **BASIC 323** ed il **BASCOM 8051**) gestiscono la linea seriale software tramite istruzioni ad alto livello.
La linea seriale B è sempre bufferata in RS 232 mentre per la seriale A è possibile selezionare, tramite una serie di comodi jumpers, il protocollo elettrico di comunicazione. In particolare può essere bufferata in RS 232, Current Loop passivo, RS 422 o RS 485; in questi ultimi casi è definibile se la comunicazione avviene in full duplex o half duplex.

ALIMENTAZIONE

L'unica tensione di alimentazione necessaria (+5 Vdc) può essere essere fornita in due modi:

- direttamente tramite i **pin 25 (GND) e 26 (+5 Vdc) del connettore CN1**
- indirettamente tramite la sezione di alimentazione switching (opzionale, codice **.SW**)

Il tipo di alimentazione della scheda non può essere variato dall'utente e deve quindi essere specificato in fase di ordine.
La tensione di alimentazione (+5 Vdc) è collegata a tutti i connettori della scheda ma per ottenere il miglior percorso della corrente si consiglia di fornirla tramite il CN1 (nella configurazione senza sezione alimentatrice) e di prelevarla da tutti gli altri connettori. Questo giustifica la direzionalità del segnale +5 Vdc riportata nella legenda di tutti i connettori. Si può comunque alimentare la scheda da un punto indicato come uscita dei +5 Vdc ma è estremamente sconsigliato e quindi eventuali conseguenze sono a carico dell'utente.
La sezione alimentatrice switching necessita di una tensione **8÷24 Vac** (12÷34 Vdc) che può essere fornita tramite appositi connettori standardizzati di facile ed immediata installazione, per maggiori informazioni vedere apposito paragrafo “TENSIONI DI ALIMENTAZIONE”.
Sulla scheda sono state adottate tutte le scelte circuitali e componentistiche che tendono a ridurre i consumi, compresa la possibilità di far lavorare il microprocessore in idle e stop mode ed a ridurre la sensibilità ai disturbi. Si ricorda inoltre che é presente una circuiteria di protezione tramite **TransZorb™** per evitare danni dovuti a tensioni non corrette.

DISPOSITIVI DI CLOCK

Sulla **GPC® 323** sono presenti due circuiti separati che provvedono a generare rispettivamente la frequenza di clock per la CPU e la frequenza per il real time clock. La scelta di utilizzare due circuiti e quindi due quarzi indipendenti, é legata alla possibilità di poter variare in modo indipendente la frequenza di lavoro della CPU senza dover effettuare altre variazioni. La frequenza per la sezione real time clock é fissa a 32768 Hz, quella per la CPU è 22,1184 MHz; da essa vengono ricavate anche le frequenze necessarie per le altre sezioni della scheda (Timer Counter, Seriali, ecc.). Si ricorda inoltre che la frequenza di clock della CPU può essere divisa via software, in modo da ridurre il consumo.

ABACO® I/O BUS

Una delle caratteristiche di fondamentale importanza della **GPC® 323** è quella di disporre del cosiddetto **ABACO®** I/O BUS: ovvero un connettore normalizzato **ABACO®** con cui è possibile collegare la scheda ad una serie di moduli esterni intelligenti e non. Tra questi si trovano moduli per acquisizione di segnali analogici (A/D), per la generazione di segnali analogici (D/A), per gestione di linee di I/O logico, per counter, ecc. e ne possono essere realizzati anche su specifiche richieste dell'utente. Utilizzando mother board come l'**ABB 03** o l'**ABB 05** é inoltre possibile gestire tutte le schede periferiche in formato Europa con interfaccia per BUS **ABACO®**. Tale caratteristica rende la scheda espandibile con un ottimo rapporto prezzo/prestazioni e quindi adatta a risolvere molti dei problemi dell'automazione industriale.

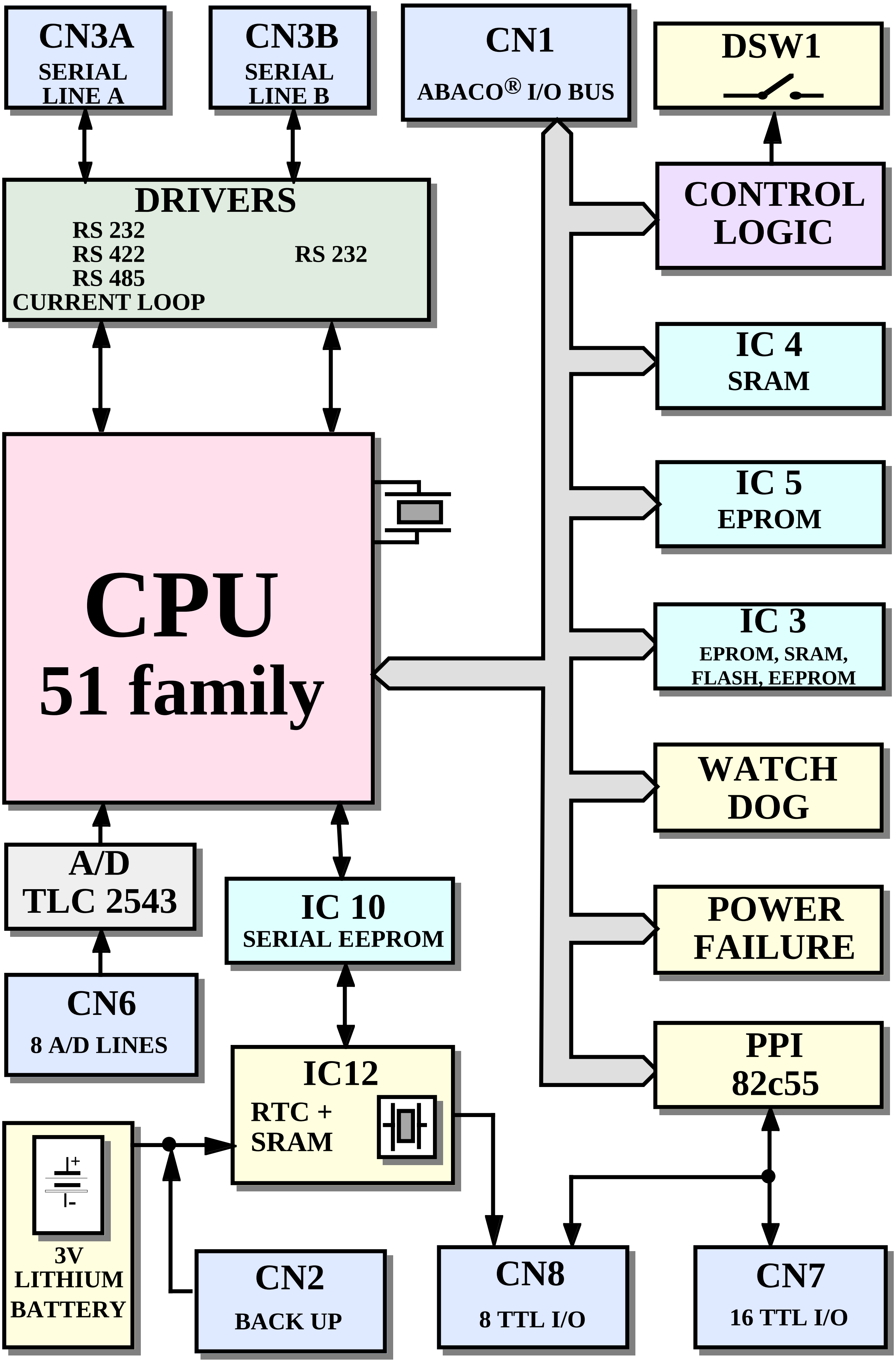


FIGURA 1: SCHEMA A BLOCCHI

PROCESSORE

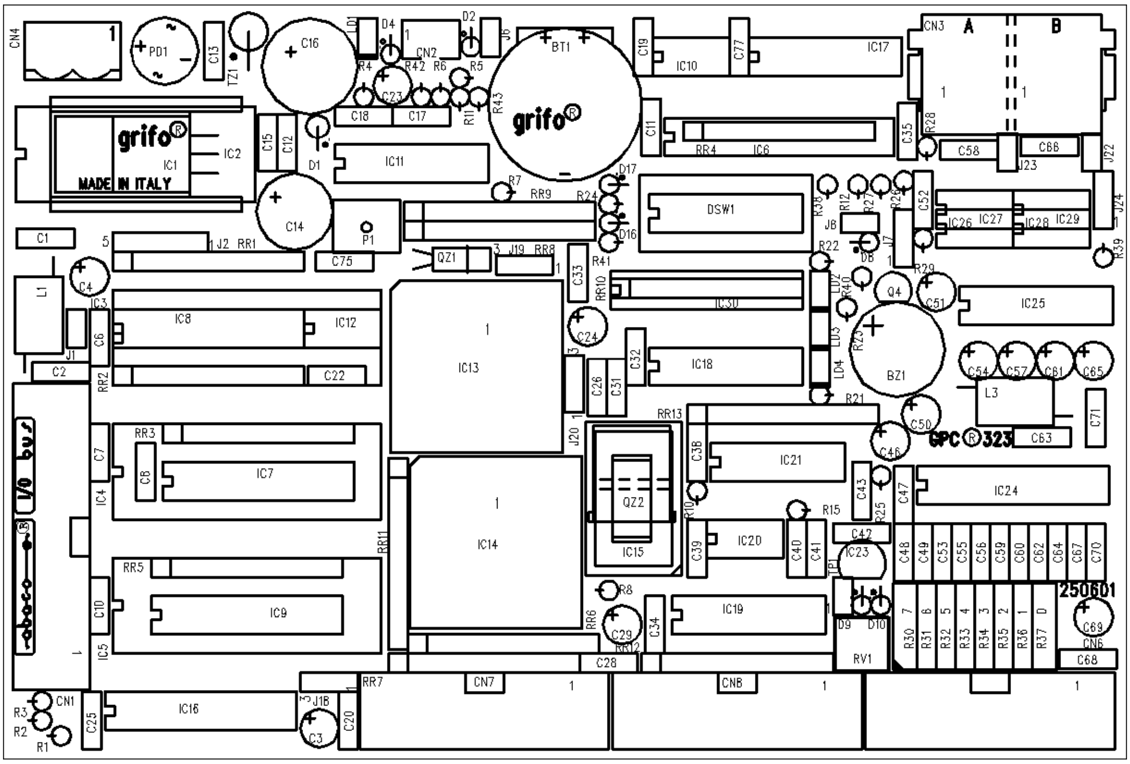
La scheda **GPC® 323** è predisposta per accettare tutti i processori con pin out compatibile con la famiglia 51 INTEL, tra questi ricordiamo: 80C32, 80C52, 87C52, 89C52 (prodotti da INTEL e numerose altre case), 89S8252 (prodotto da ATMEL), 89CRx+/2 (prodotti da PHILIPS), 80C320,87C320 (prodotti da DALLAS). Tali processori ad 8 bits sono codice compatibile con la famiglia INTEL 8051, largamente diffusa a livello mondiale, e sono caratterizzati da: un esteso set di istruzioni, un'alta velocità di esecuzone e di manipolazione dati, da un efficiente gestione degli interrupts e da una ricca serie di periferiche harware integrate. In questo manuale in tutti i paragrafi viene riportata una descrizione delle caratteristiche comuni a tutti i microprocessori effettuando, quando necessario, le dovute distinzioni.
Di seguito viene riportato un elenco delle caratteristiche principali delle CPU disponibili:

Microprocessore	**80C32**	**89S8252**	**89CRx+/2**	**80C320**
Dimensione BUS dati	8	8	8	8
Clock per ciclo macchina	12	12	6	4
RAM interna (bytes)	256	256	256	256
ROM interna (kbytes)	8	8	64	8
EEPROM interna (kbytes)	0	2	0	0
Area codice esterna (kbytes)	64	64	64	64
Area dati esterna (kbytes)	64	64	64	64
Ports di I/O	4	4	4	4
Timer/Counters a 16 bits	3	3	3	3
Sorgenti d'nterrupt	6	9	7	13
Livelli di priorità interrupt	2	2	4	3
Linee seriali a/sincrone	1	1	1	2
Modalità a basso consumo	Si	Si	Si	Si
Gestione consumo e controllo	No	Si	No	Si
Watch dog timer interno	No	Si	No	Si
Programmazione su scheda	No	Si	Si	No
Programmazione da applicativo	No	No	Si	No

Per maggiori informazioni si faccia riferimento all'apposita documentazione della casa costruttrice o all'appendice B di questo manuale. Si ricorda che la precedente tabella descrive le caratteristiche generali dei microprocessori ed alcune di queste possono non essere supportate dalla scheda.
L'utente deve specificare il microprocessore richiesto durante la fase di ordine ed in caso di assenza di indicazioni la scheda é fornita nella sua versione base con 80C32. La versione con il DALLAS 80C320 é invece indicata con il suffisso D, ovvero **GPC® 323D**.

LOGICA DI CONTROLLO

Il mappaggio di tutti i registri delle periferiche presenti sulla scheda e dei dispositivi di memoria, è affidata ad un'opportuna logica di controllo, realizzata con una logica programmabile, che si occupa di allocare tali dispositivi nello spazio d'indirizzamento della CPU. Per maggiori informazioni fare riferimento al capitolo MAPPAGGI ED INDIRIZZAMENTI.



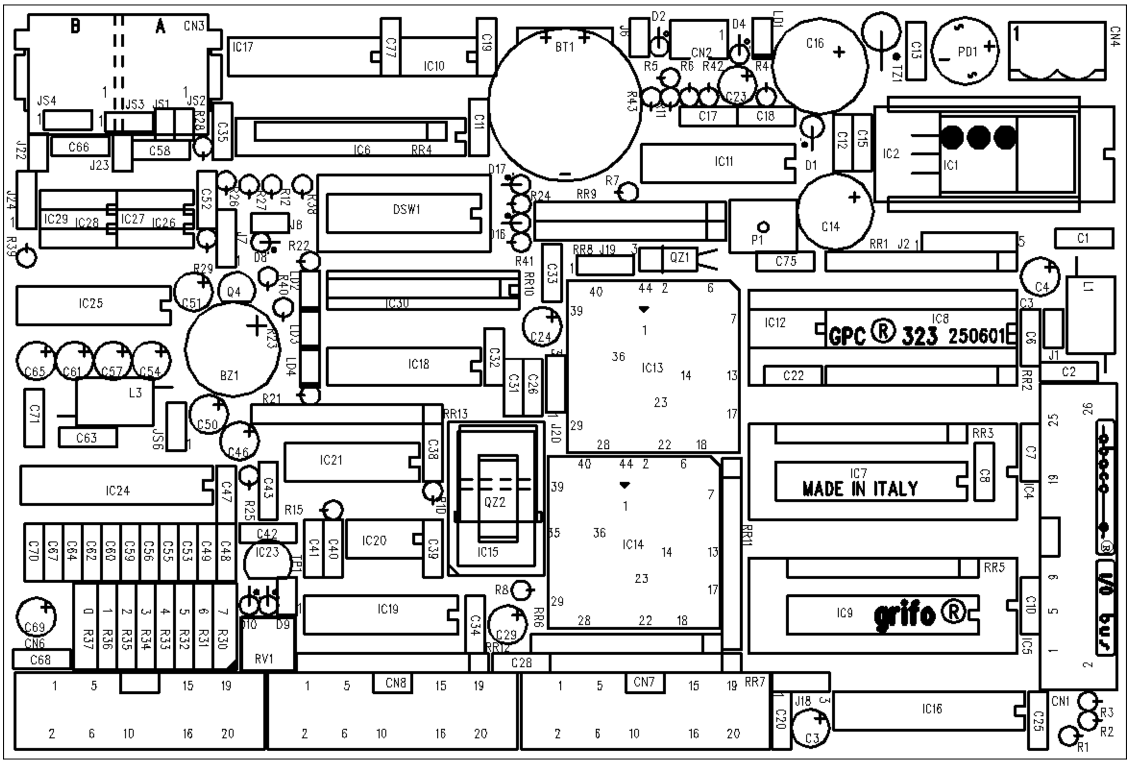


FIGURA 2: MAPPA DEI COMPONENTI

DISPOSITIVI PERIFERICI DI BORDO

La scheda **GPC® 323**, nata per risolvere molteplici problemi di controllo e comando di automatismi, è dotata di alcuni componenti periferici che si occupano dell'interfacciamento con il mondo esterno. In particolare:

- **Watch dog esterno**: circuiteria di tipo astabile in grado di resettare la scheda ad intervallo di tempo di circa 1,4 sec. Via software l'utente é in grado di retriggerare la circuiteria per evitare il reset tramite l'acceso ad opportuni registri situati nello spazio d'indirizzamento della CPU. Nel caso in cui la scheda utilizzi microprocessori provvisti di sezione di watch dog interna l'utente può disporre di due circuiterie separate con caratteristiche diverse e quindi con un livello di sicurezza superiore. e conferisce al sistema basato sulla scheda, una sicurezza estrema
- **EEPROM seriale**: il modulo di EEPROM seriale (IC10) è molto utile in caso si debbano mantenere delle informazioni anche in assenza di alimentazione, senza ricorrere al back up della SRAM, con una sicurezza estrema sulla validità dei dati. Tale modulo può avere un size che varia nel range 512÷2048 bytes e di default viene montato un modulo con un size di 512 bytes.
- **Configurazione scheda**:allo scopo di rendere configurabile la scheda ed in particolare il programma applicativo sviluppato é stato previsto un dip switch a 8 vie, di cui 5 switches sono acquisibili dall'utente. La possibilità di acquisire via software lo stato di questi switches fornisce all'utente la possibilità di gestire diverse condizioni tramite un unico programma, senza dover rinunciare ad altre linee d'ingresso (le applicazioni caratteristiche sono: selezione della lingua di rappresentazione, definizione parametri del programma, selezione delle modalità operative, ecc.). Sempre in merito alla configurazione della scheda, sulla **GPC® 323** sono stati previsti due LED di attività, gestiti via software con cui l'utente può segnalare visivamente lo stato di tutto il sistema.
- **Real Time Clock**: il modulo di SRAM tamponata installato in IC 12 è provvisto di un completo Real Time Clock in grado di gestire ore, minuti, secondi, giorno del mese, mese, anno e giorno della settimana in modo completamente autonomo, essendo tamponato il riferimento temporale a bordo scheda diviene estremamente sicuro ed affidabile. Il componente é inoltre in grado di generare un segnale di interrupt con cadenza programmabile tra varie opzioni.
- **Linee di I/O**: sulla scheda sono presenti tre port paralleli da 8 bit per un totale di 24 linee di I/O digitale a livello TTL, con direzionalità settabile a livello di port, gestite dal PPI 82C55. Tali linee sono collegate direttamente a due connettori standardizzati di I/O **ABACO®** ed hanno la possibilità di essere direttamente collegate a numero schede d'interfaccia. Le 24 linee di I/O vengono completamente gestite via software tramite la programmazione di 4 registri situati nello spazio di I/O della CPU da un'apposita logica di controllo.
- **A/D Converter**:la sezione di A/D converter della **GPC® 323** é basata su un convertitore in grado di acquisire 11 canali con una risoluzione massima di 12 bits. Dal punto di vista software é possibile definire quali canali attivare, dare lo start o lo stop all' acquisizione ecc., tramite la gestione di una comunicazione sincrona con il dispositivo. Al fine di semplificare la gestione dello stesso A/D alcuni pacchetti software forniscono delle procedure di utility che gestiscono la sezione in tutte le sue parti. I segnali analogici collegabili sono segnali in tensione variabili nel range 0÷2,490 V, 0÷5,000 V (opzione **.VREF-5**) oppure in corrente variabili nel range 0÷20 mA (opzione **.8420**); possono essere richieste anche delle configurazioni miste con alcuni ingressi analogici in tensioni ed alcuni in corrente. La sezione di A/D converter é opzionale e deve essere quindi esplicitamente ordinata (codice **.AD**), specificando anche il tipo degli ingressi analogici.
- **Buzzer**:sulla **GPC® 323** è presente una circuiteria in grado di emettere un suono costante, basata su un buzzer capacitivo. Questa circuiteria viene abilitata e/o disabilitata via software tramite la logica di controllo e può essere utilizzata per generare allarmi acustici, feed back sonori, ecc.

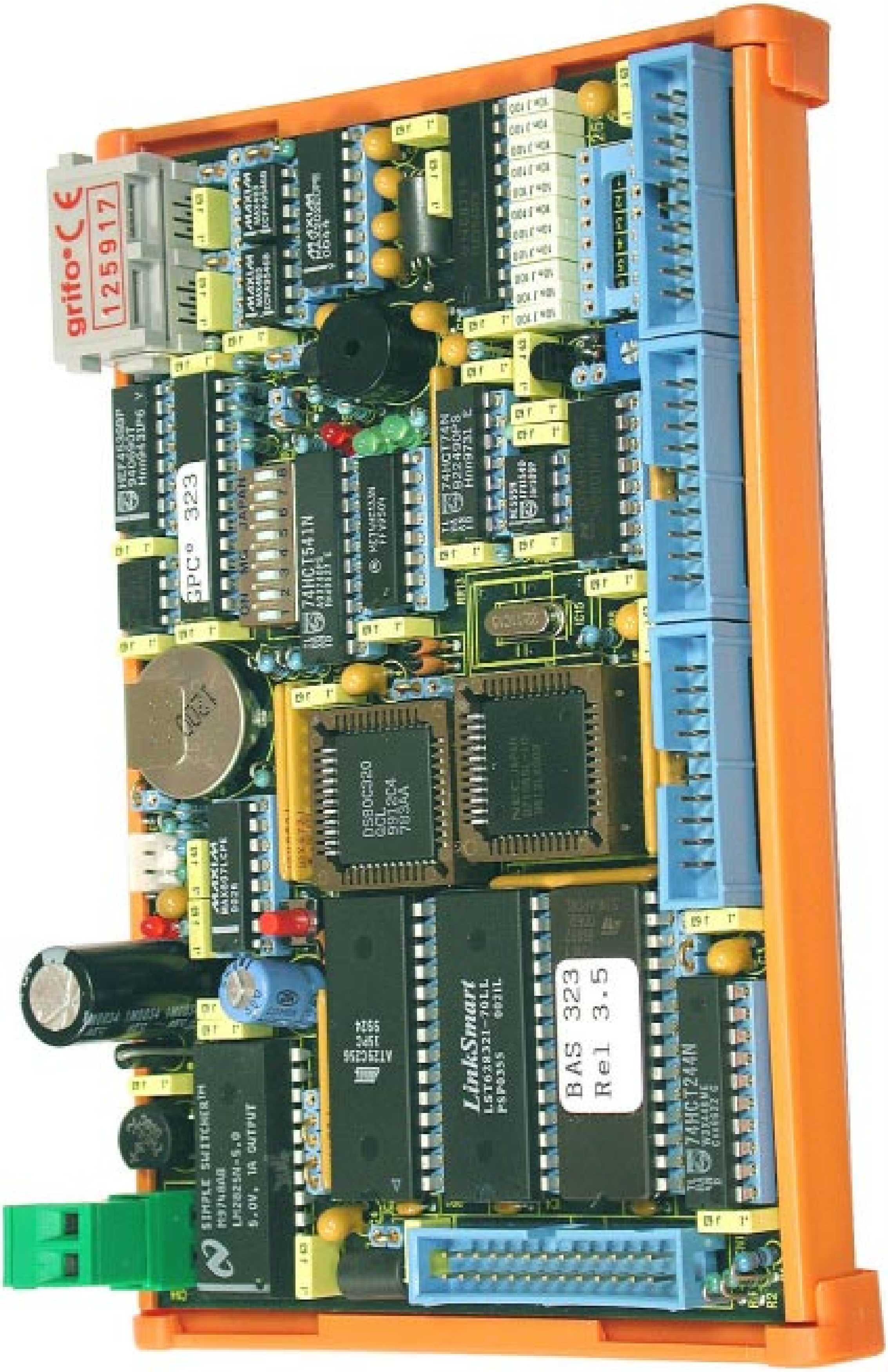


FIGURA 3: FOTO SCHEDA

SPECIFICHE TECNICHE

CARATTERISTICHE GENERALI

Risorse di bordo:	24 input/output programmabili TTL 1 linea bidirezionale RS 232, RS 422, RS 485 o Current Loop 1 linea bidirezionale RS 232 (hardware/software) 3 timer counter 1 buzzer 1 dip switch a 8 vie 2 LEDs utilizzabili dall'utente 1 watch dog 1 contatto locale di reset 1 ingresso configurazione leggibile da software 1 interfaccia di espansione **ABACO® I/O BUS** 1 orologio tamponato 1 circuiteria di power failure
Memoria indirizzabile:	IC 5: EPROM da 32K x 8 IC 4: SRAM da 32K x 8 IC 3: SRAM,EPROM,EEPROM,FLASH EPROM da 32K x 8 IC 10:EEPROM seriale da 256 bytes a 2048 bytes IC 12:SRAM+RTC da 256 bytes
Tempo di accesso memorie:	70 nsec
CPU di bordo:	INTEL 80C32 e compatibili ATMEL 89S8252 e compatibili PHILIPS 89CRx+/2 e compatibili DALLAS 80C320 e compatibili
Frequenza quarzo (clock):	22,1184 MHz
Risoluzione A/D:	12 bit
Tempo conversione A/D:	10 µsec
Tempo intervento watch dog:	da 940 msec a 2060 msec (tipico 1420 msec)

CARATTERISTICHE FISICHE

Dimensioni (L x A x P):	100 x 149 x 25 mm	
Peso:	178 g	
Range di temperatura:	da 0 a 50 gradi Centigradi	
Umidità relativa:	20% fino a 90%	(senza condensa)

Connettori:	CN1:	26 vie scatolino verticale M
	CN2:	2 vie scatolino verticale M
	CN3A:	PLUG a 6 vie
	CN3B:	PLUG a 6 vie
	CN4:	2 vie morsettiera
	CN6:	20 vie scatolino verticale M
	CN7:	20 vie scatolino verticale M
	CN8:	20 vie scatolino verticale M

CARATTERISTICHE ELETTRICHE

Tensione di alimentazione:	5 Vdc		(senza sezione alimentatrice)
	6÷12 Vac	(9÷16 Vdc)	(sezione alimentatrice lineare)
	8÷24 Vac	(12÷34 Vdc)	(sezione alimentatrice switching)
Corrente assorbita sui +5 Vdc:	156 mA		(configurazione base)
	199 mA		(conf. massima)
Corrente fornita sui +5 Vdc per carichi esterni:	Dipende dall'alimentatore	*	(senza sezione alimentatrice)
	800 mA	*	(con sezione alimentatrice)
Batteria di backup:	3.0 Vdc; 180 mAh		
Batteria esterna di back up:	3,6÷5 Vdc		
Corrente di back up:	4.2 μA (batteria di bordo)		
	5.2 μA (batteria esterna da 3.6 V)		
Ingressi analogici in tensione:	0÷2,49 V oppure 0÷5,00 V		(opzionale)
Ingressi analogici in corrente:	0÷20; 4÷20 mA		(con modulo di conversione)
Impedenza ingressi analogici:	elevata		
Rete terminazione RS 422-485:	Resistenza terminazione linea=		120 Ω
	Resistenza di pull up sul positivo=		3,3 KΩ
	Resistenza di pull down sul negativo=		3,3 KΩ
Soglia del power failure:	52 mV prima dell'attivazione del reset		

* I dati riportati sono riferiti ad un lavoro a temperatura ambiente di 20 gradi centigradi (per ulteriori informazioni fare riferimento al paragrafo “TENSIONE DI ALIMENTAZIONE”).

INSTALLAZIONE

In questo capitolo saranno illustrate tutte le operazioni da effettuare per il corretto utilizzo della scheda. A questo scopo viene riportata l'ubicazione e la funzione degli strip, dei connettori e dei LEDs, ecc. presenti sulla **GPC® 323**.

CONNESSIONI CON IL MONDO ESTERNO

Il modulo **GPC® 323** è provvisto di 7 connettori con cui vengono effettuati tutti i collegamenti con il campo e con le altre schede del sistema di controllo da realizzare. Di seguito viene riportato il loro pin out ed il significato dei segnali collegati; per una facile individuazione di tali connettori, si faccia riferimento alla figura 9, mentre per ulteriori informazioni a riguardo del tipo di connessioni, fare riferimento alle figure successive che illustrano il tipo di collegamento effettuato a bordo scheda.

CN4 - CONNETTORE ALIMENTAZIONE CON SEZIONE ALIMENTATRICE

CN4 é un connettore a morsettiera per rapida estrazione, composto da 2 contatti. Tramite CN4 deve essere fornita la tensione di alimentazione della scheda. Utilizzando la scheda senza la sezione alimentatrice, la tensione +5 Vdc deve essere fornita tramite il pin 26 (+Vdc) e il pin 25 (GND) di CN1.

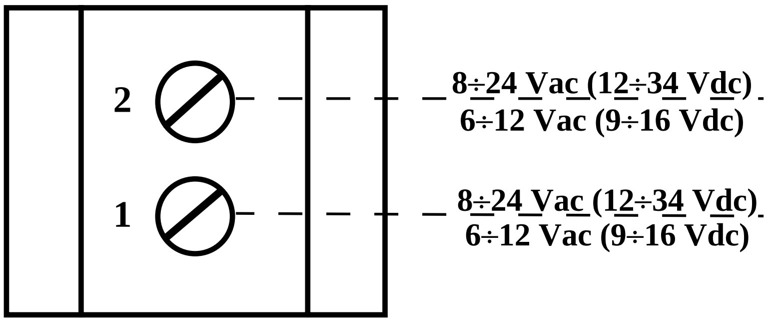


FIGURA 4: CN4 - CONNETTORE ALIMENTAZIONE CON SEZIONE ALIMENTATRICE

Legenda:

8÷24 Vac (12÷34 Vdc) = I- In gresso alimentazione 8÷24 Vac (12÷34 Vdc) (sezione switching)
6÷12 Vac (9÷16 Vdc) oppure 6÷12 Vac (9÷16 Vdc) (sezione lineare)

CN1 - CONNETTORE PER ABACO® I/O BUS

CN1 è un connettore a scatolino verticale con passo 2.54 mm a 26 piedini. Tramite CN1 si effettua la connessione tra la scheda e la serie di moduli esterni di espansione, da utilizzare per l'interfacciamento diretto con il campo. Tale collegamento è effettuato tramite l'**ABACO®** I/O BUS di cui questo connettore riporta tutti i segnali a livello TTL.

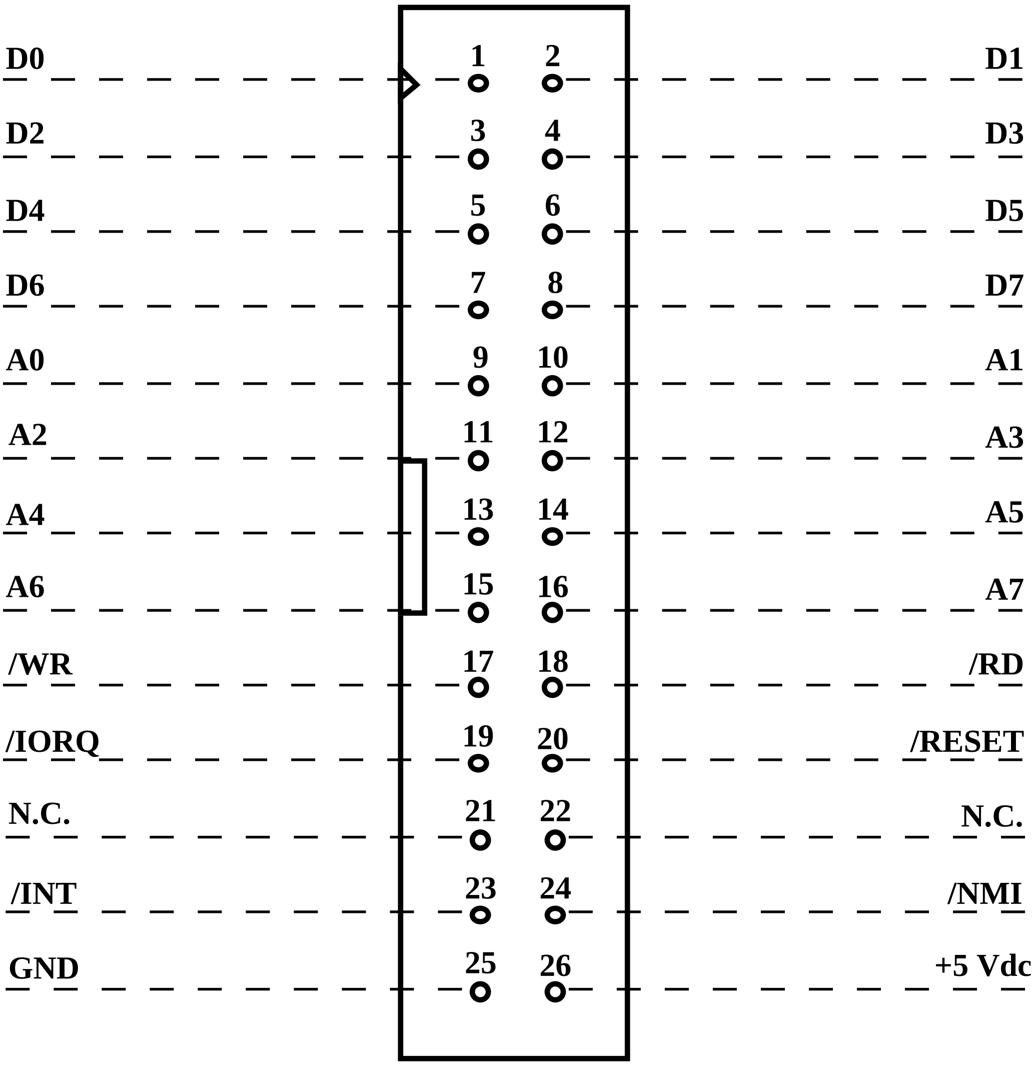


FIGURA 5: CN1 - CONNETTORE PER ABACO® I/O BUS

Legenda:

A0-A7	=	O	- Address BUS: BUS degli indirizzi.
D0-D7	=	I/O	- Data BUS: BUS dei dati.
/INT BUS	=	I	- Interrupt request: richiesta d'interrupt. Deve essere in open collector
/NMI BUS	=	I	- Non Mascable Interrupt: richiesta d'interrupt non mascherabile.
/IORQ	=	O	- Input Output Request: richiesta operazione Input Output su I/O BUS.
/RD	=	O	- Read cycle status: richiesta di lettura.
/WR	=	O	- Write cycle status: richiesta di scrittura.
/RESET	=	O	- Reset: azzeramento.
/CS1, /CS2	=	O	- Chip select 1 e 2: abilitazione decodificata per periferiche esterne
+5 Vdc	=	I	- Linea di alimentazione a +5 Vdc.
GND	=		- Linea di massa.

CN7 - CONNETTORE PER I/O DEL PPI 82C55, PORT A E C

CN7 è un connettore a scatolino verticale con passo 2.54 mm a 20 piedini. Tramite CN7 si effettua la connessione tra i port A e C della periferica programmabile PPI 82C55 e l'ambiente esterno utilizzando due dei tre port paralleli ad 8 bit. I segnali presenti su questo connettore coincidono con segnali logici a livello TTL e seguono il pin out standardizzato **I/O ABACO®**.

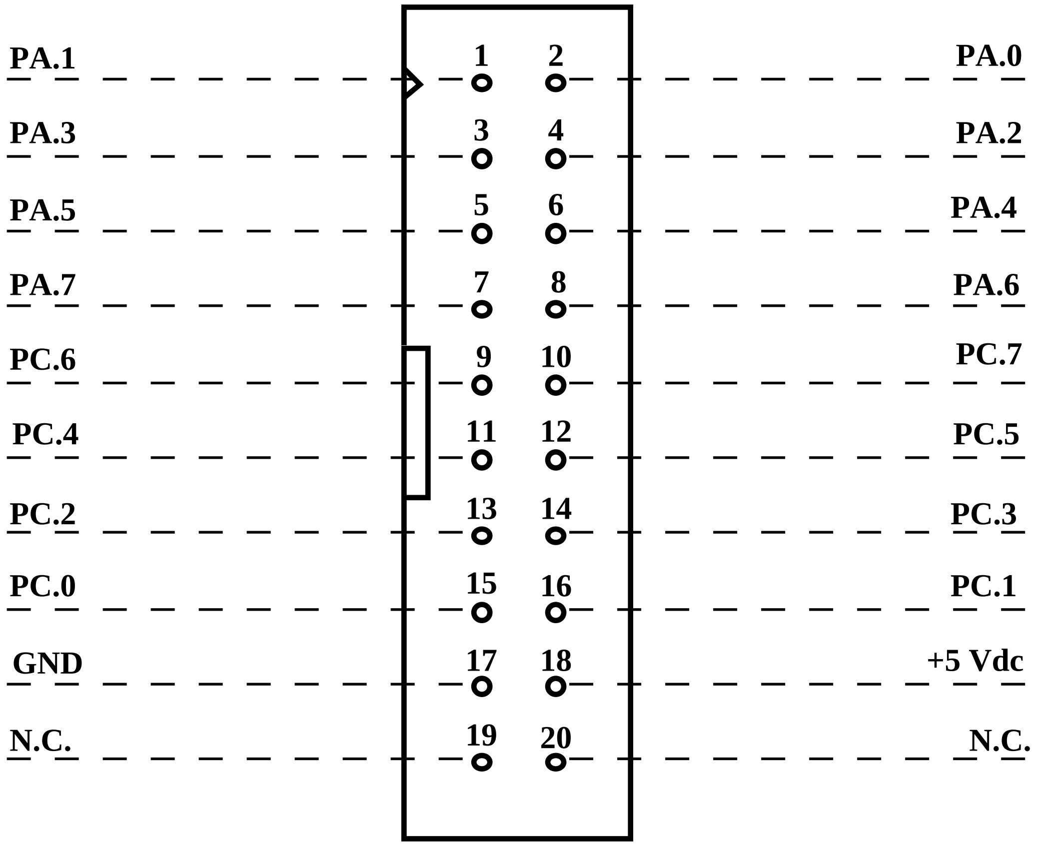


FIGURA 6: CN7 - CONNETTORE PER I/O DEL PPI 82C55, PORT A E C

Legenda:

PA.n	=	I/O	-	Linea digitale n del port A del PPI 82C55.
PC.n	=	I/O	-	Linea digitale n del port C del PPI 82C55.
GND	=		-	Linea di massa.
+5 Vdc	=	O	-	Linea di alimentazione a +5 Vdc.
N.C.	=		-	Non Collegato.

Di particolare interesse é la possibilità di collegare direttamente al connettore CN7 una serie di interfacce con cui risolvere molti problemi caratteristici del settore industriale. Per i moduli di interfaccia operatore locale (**QTP 24P**, **KDL x24**, **KDF 224**, ecc.), oltre alla facilità d'installazione si ricorda anche la facilità di gestione software, supportata da linguaggi di programmazione ad alto livello. Per maggiori informazioni si faccia riferimento al paragrafo "INTERFACCE OPERATORE LOCALI".

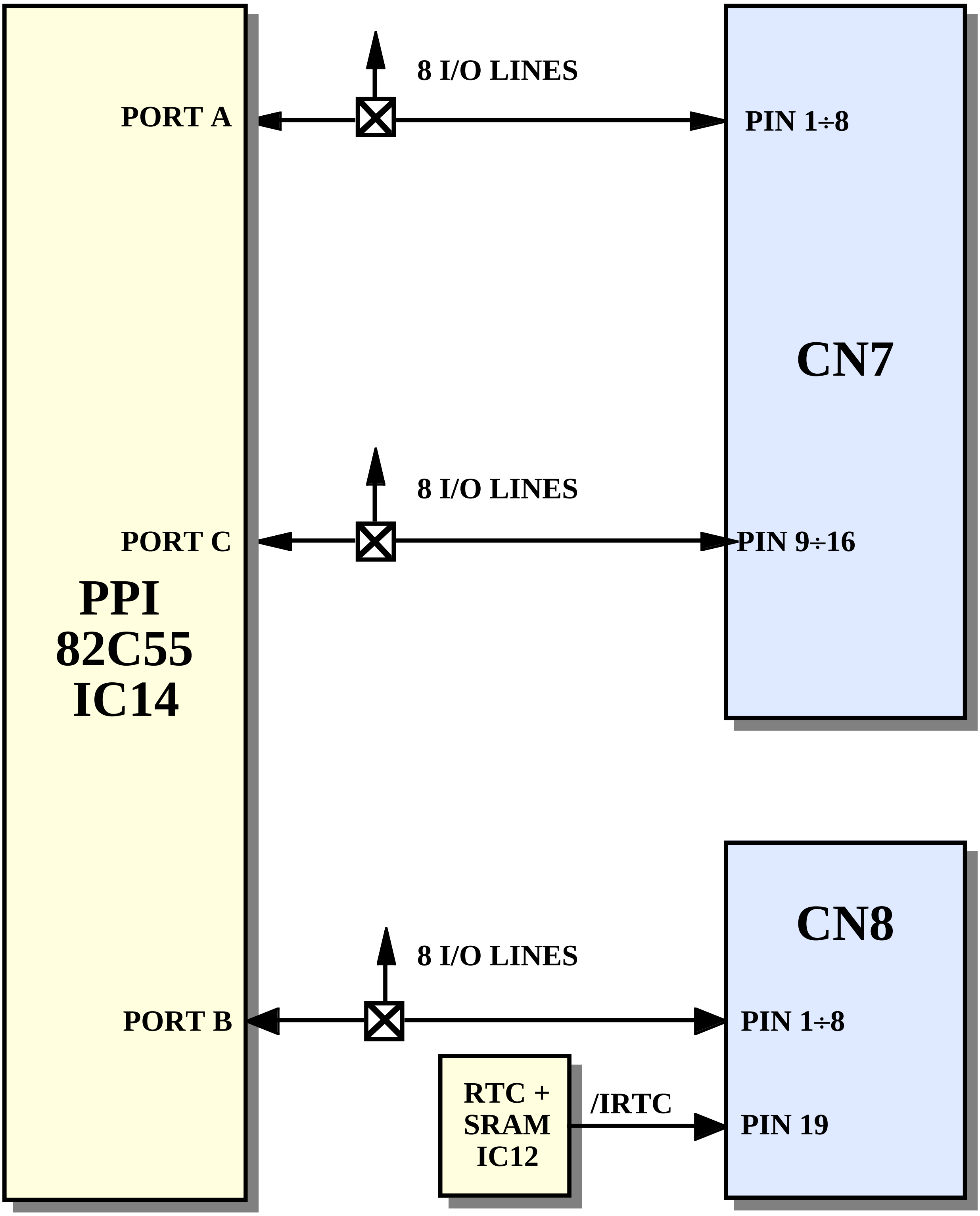


FIGURA 7: SCHEMA COLLEGAMENTO LINEE DI I/O

CN8 - CONNETTORE PER I/O DEL PPI 82C55, PORT B E INTERRUPT RTC

CN8 è un connettore a scatolino verticale con passo 2.54 mm a 20 piedini. Tramite CN8 si effettua la connessione tra il port B della periferica programmabile PPI 82C55 e l'ambiente esterno utilizzando uno dei tre port paralleli ad 8 bit. I segnali presenti su questo connettore coincidono con segnali logici a livello TTL e seguono il pin out standardizzato **I/O ABACO®**.
Inoltre è presente il segnale di interrupt periodico generato dall'RTC. Tale segnale può essere disabilitato mentre, se abilitato, il periodo può essere scelto tra varie opzioni.

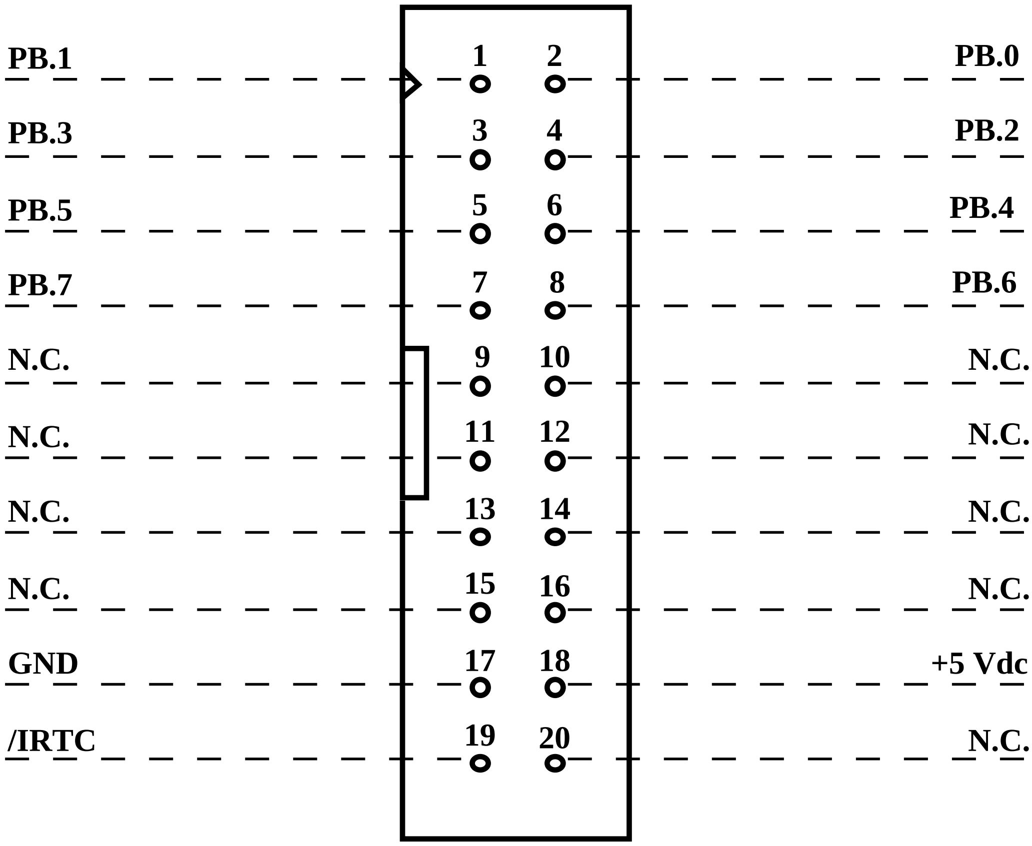


FIGURA 8: CN8 - CONNETTORE PER I/O DEL PPI 82C55, PORT B E INTERRUPT RTC

Legenda:

PB.n	=	I/O	-	Linea digitale n del port B del PPI 82C55.
GND	=		-	Linea di massa.
/IRTC	=	O	-	Interrupt periodico generato dal Real Time Clock
+5 Vdc	=	O	-	Linea di alimentazione a +5 Vdc.
N.C.	=		-	Non Collegato.

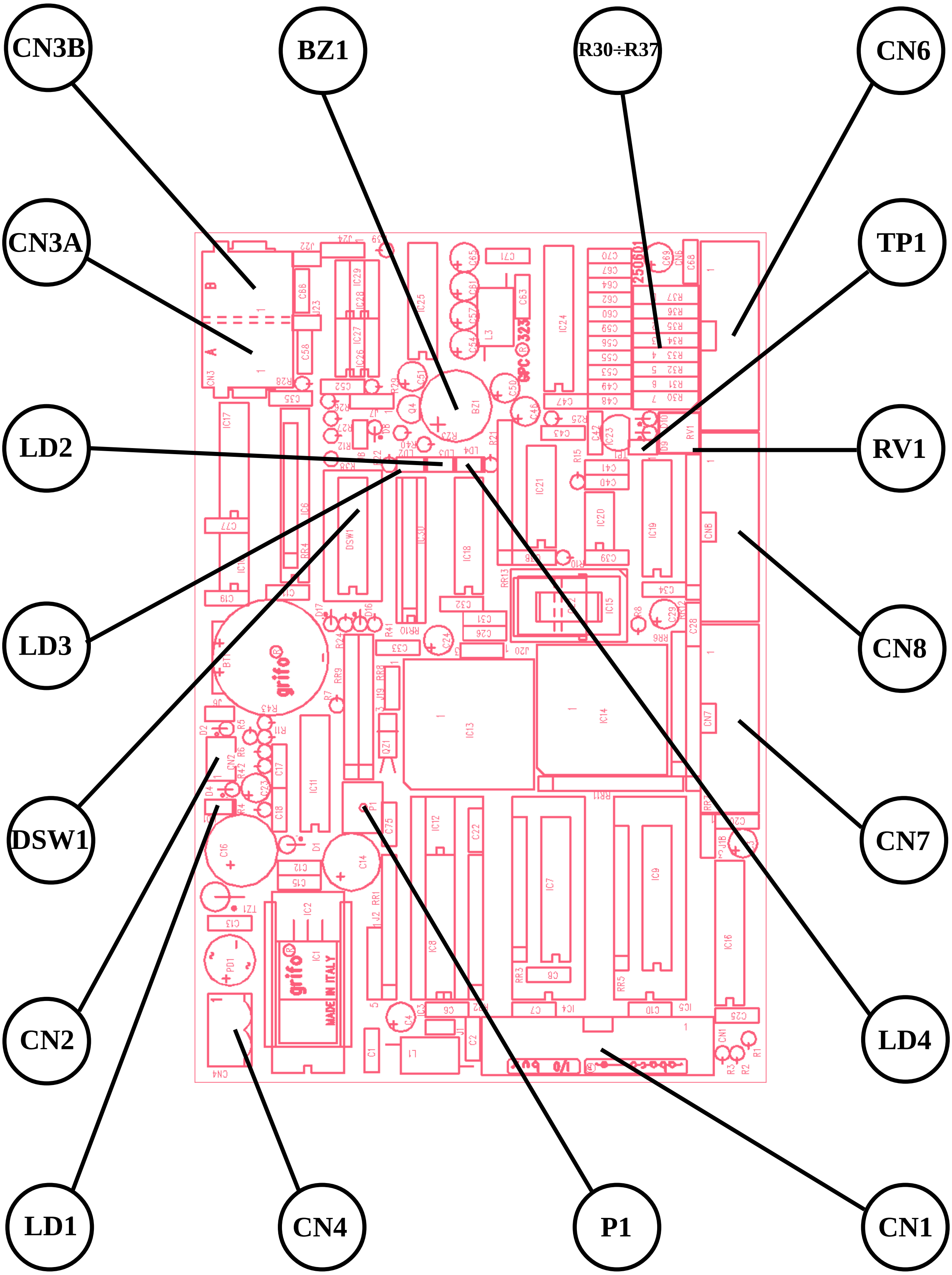


FIGURA 9: DISPOSIZIONE LEDS, CONNETTORI, DIP SWITCH, ECC.

CN3A - CONNETTORE PER LINEA SERIALE A

CN3A è un connettore femmina, del tipo PLUG a 6 vie.
Sul connettore sono disponibili i segnali per la comunicazione della linea seriale A in RS 232, RS 422,RS 485 o Current Loop che é fisicamente collegato alla seriale hardware 0 del microprocessore. La disposizione di tali segnali, riportata di seguito, è stata studiata in modo da ridurre al minimo le interferenze ed in modo da facilitare la connessione con il campo, mentre i segnali rispettano le normative definite dal CCITT relative ad ognuno degli standard di comunicazione usati.

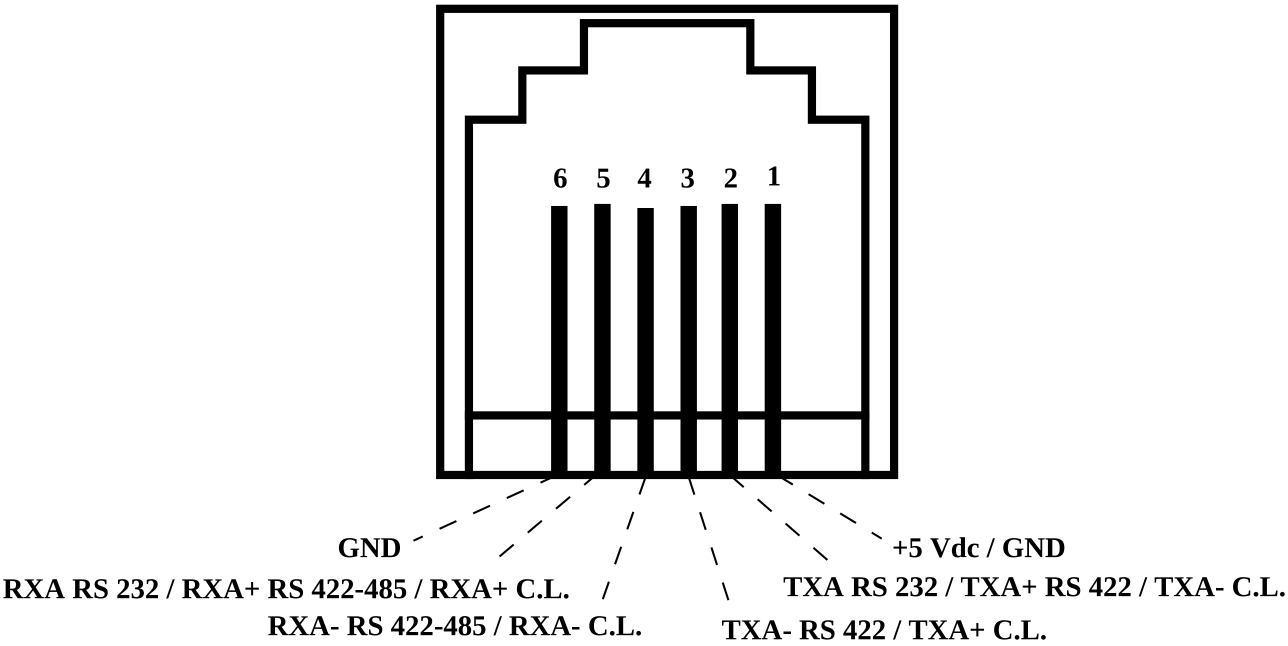


FIGURA 10: CN3A-CONNETTORE PER LINEA SERIALE A

Legenda:

RXA RS232	= I - Receive Data: linea ricezione in RS 232 della seriale A.
TXA RS232	= O - Transmit Data: linea trasmissione in RS 232 della seriale A.
RXA- RS422	= I - Receive Data Negative: linea bipolare negativa di ricezione differenziale in RS 422 della seriale A.
RXA+ RS422	= I - Receive Data Positive: linea bipolare positiva di ricezione differenziale in RS 422 della seriale A.
TXA- RS422	= O - Transmit Data Negative: linea bipolare negativa di trasmissione differenziale in RS 422 della seriale A.
TXA+ RS422	= O - Transmit Data Positive: linea bipolare positiva di trasmissione differenziale in RS 422 della seriale A.
RXTXA- RS485	=I/O- Receive Transmit Data Negative: linea bipolare negativa di ricezione e trasmissione differenziale in RS 485 della seriale A.
RXTXA+ RS485	=I/O- Receive Transmit Data Positive: linea bipolare positiva di ricezione e trasmissione differenziale in RS 485 della seriale A.
RXA- C.L.	= I - Receive Data Negative: linea bipolare negativa di ricezione in Current Loop della seriale A.
RXA+ C.L.	= I - Receive Data Positive: linea bipolare positiva di ricezione in Current Loop della seriale A.
TXA- C.L.	= O - Transmit Data Negative: linea bipolare negativa di trasmissione in Current Loop della seriale A.

TXA+ C.L. = O - Transmit Data Positive: linea bipolare positiva di trasmissione in Current Loop della seriale A.
+5 Vdc = O - Linea di alimentazione a +5 Vdc.
GND = - Linea di massa.

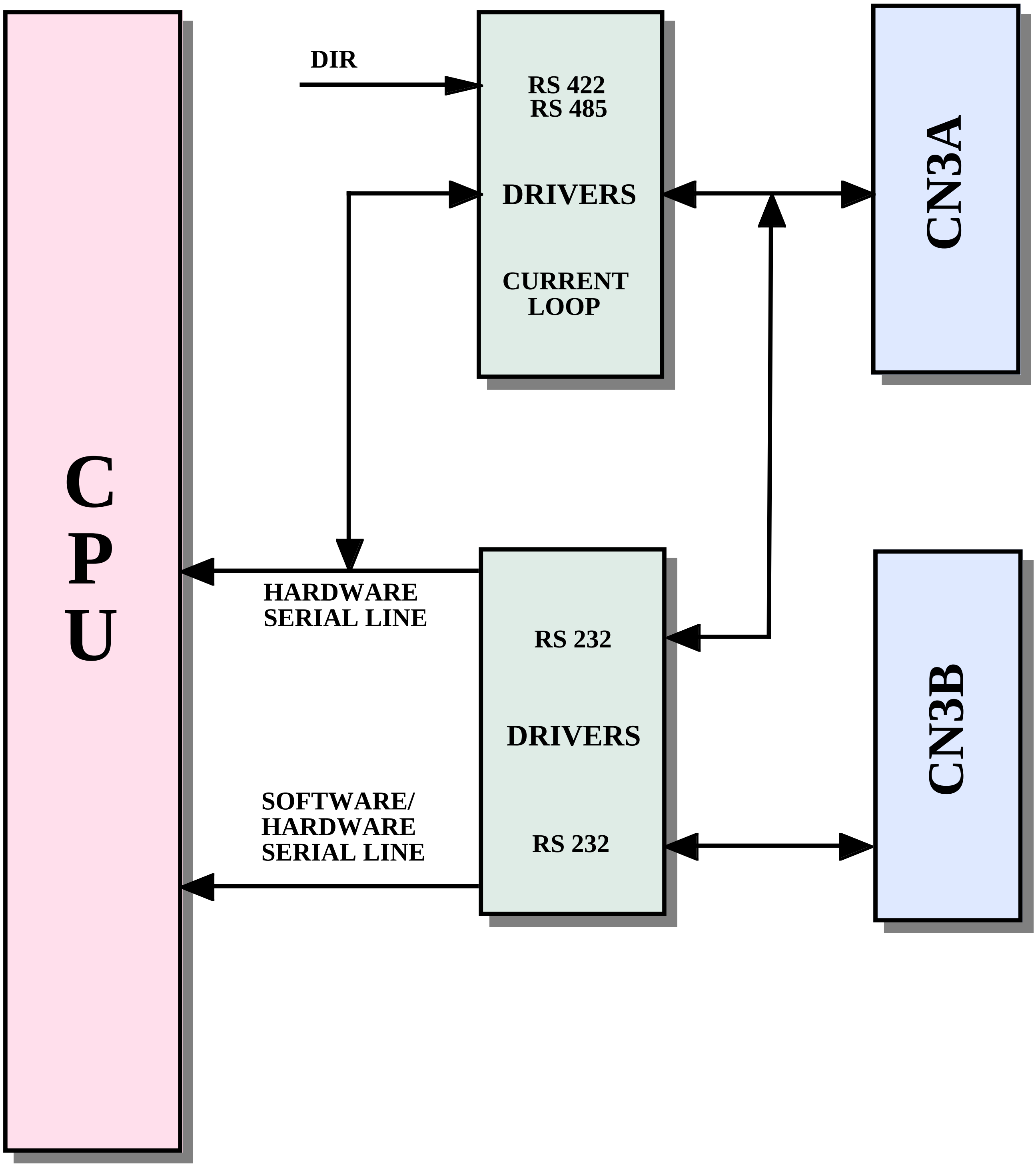


FIGURA 11: SCHEMA DI COMUNICAZIONE SERIALE

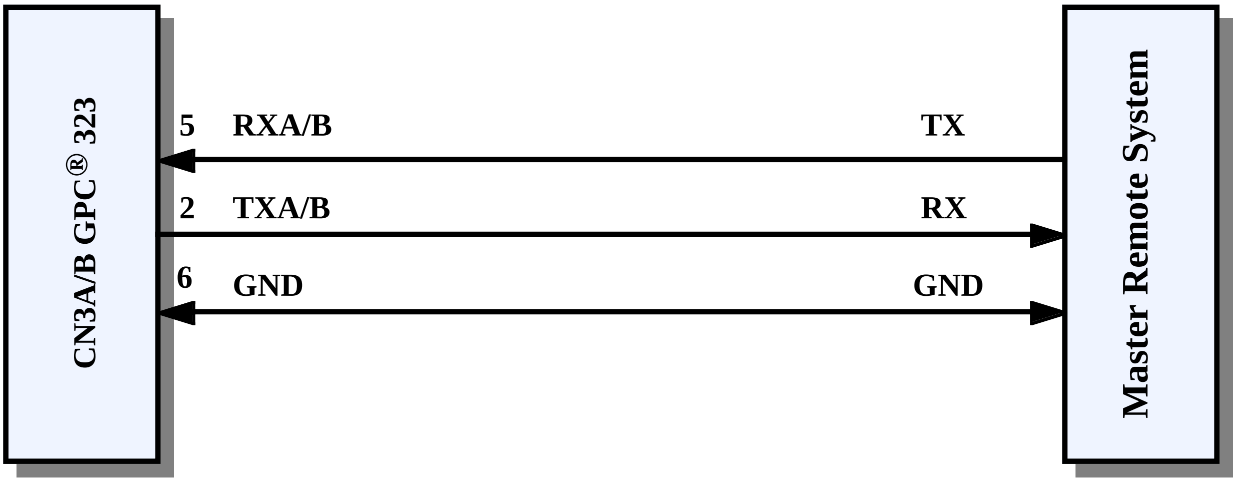


FIGURA 12: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN RS 232

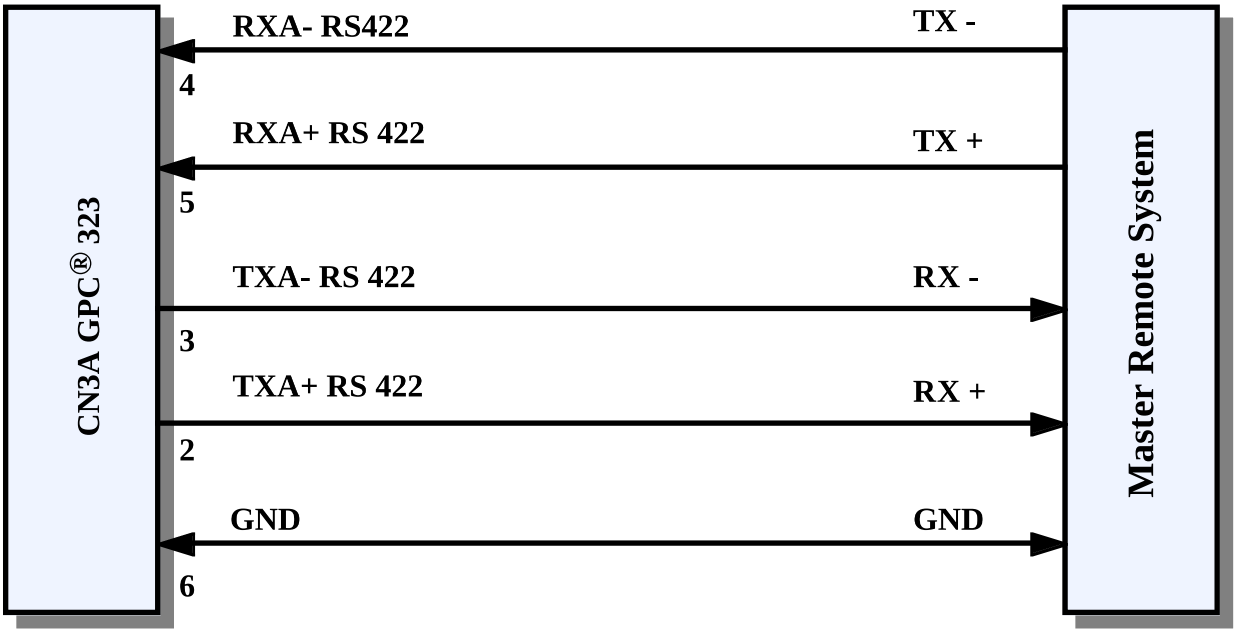


FIGURA 13: ESEMIPO COLLEGAMENTO PUNTO PUNTO IN RS 422

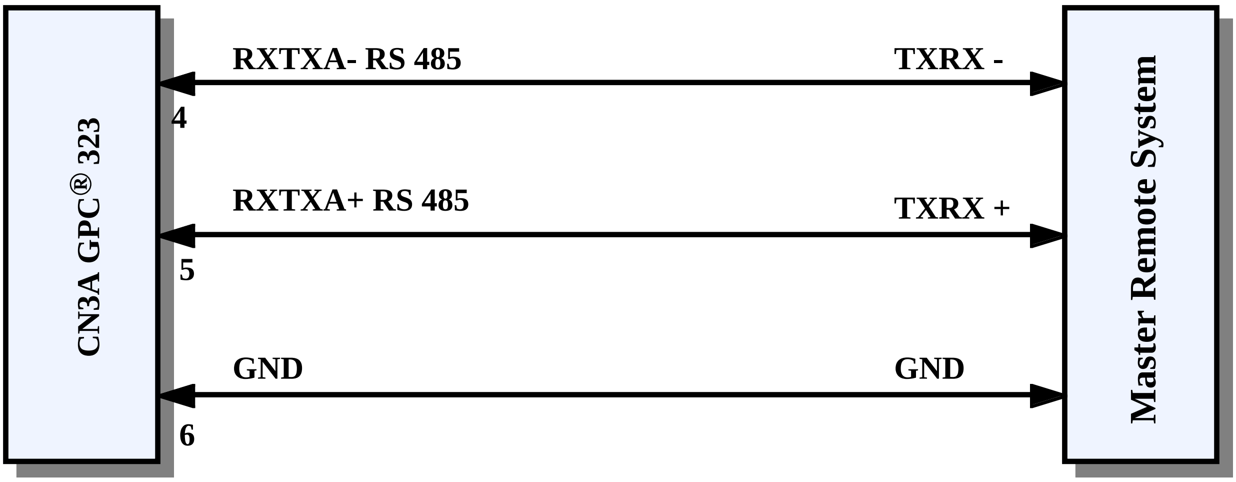


FIGURA 14: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN RS 485

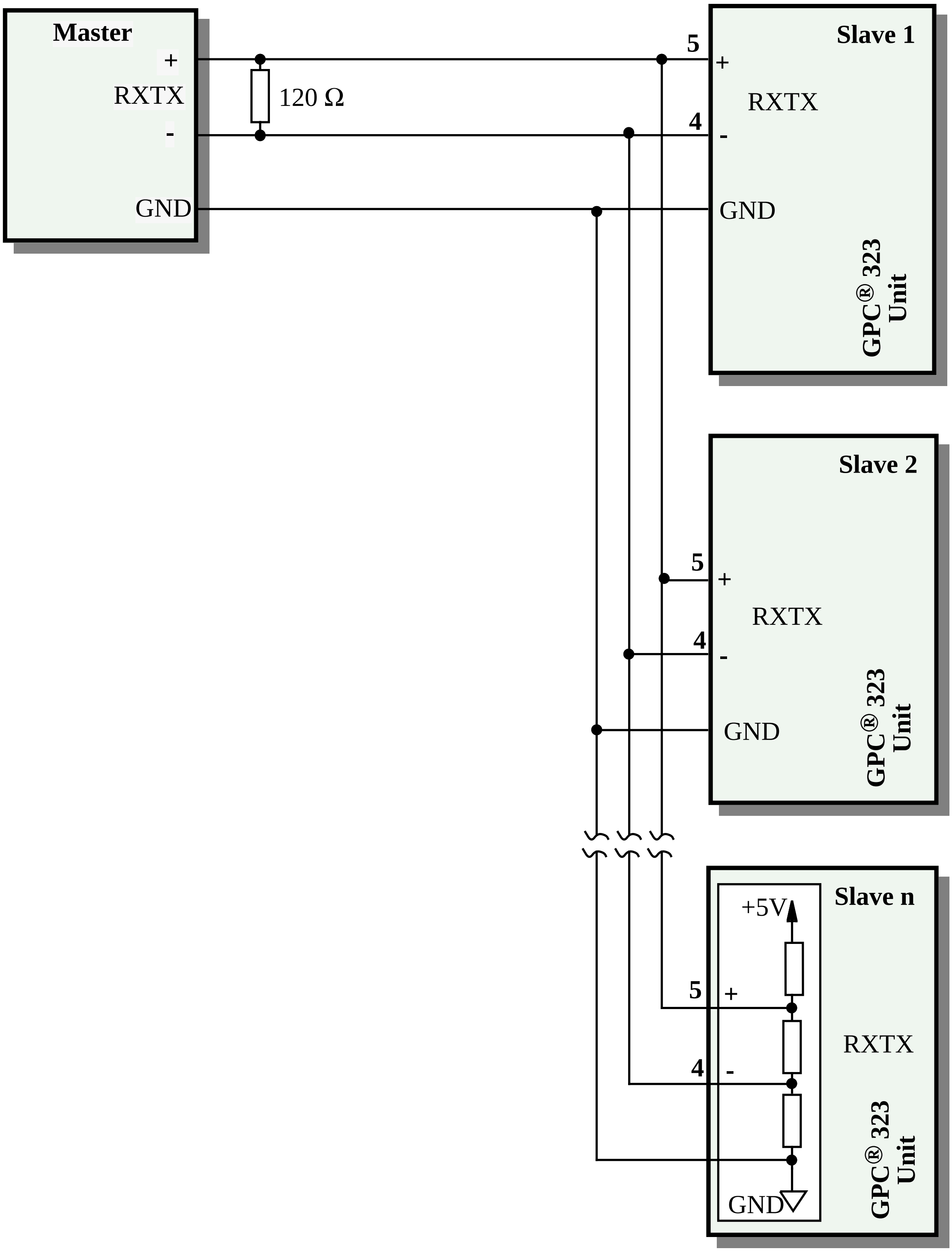


FIGURA 15: ESEMPIO COLLEGAMENTO IN RETE IN RS 485

Da notare che in una rete RS 485, devono essere presenti due resistenze di forzatura lungo la linea e due resitenze di terminazione (120 Ω), alle estremità della stessa, rispettivamente vicino all'unità Master ed all'ultima unità Slave.
A bordo della **GPC® 323** è presente la circuiteria di terminazione e forzatura, che può essere inserita o disinserita, tramite appositi jumpers, come illutrato in seguito.
In merito alla resistenza di terminazione dell'unità Master, provvedere a collegarla solo se questa non é già presente al suo interno (ad esempio molti convertitori RS232-RS485 ne sono già provvisti).
Per maggiori informazioni consultare il Data-Book TEXAS INSTRUMENTS, "*RS 422 and RS 485 Interface Cicuits*", nella parte introduttiva riguardante le reti RS 422-485.

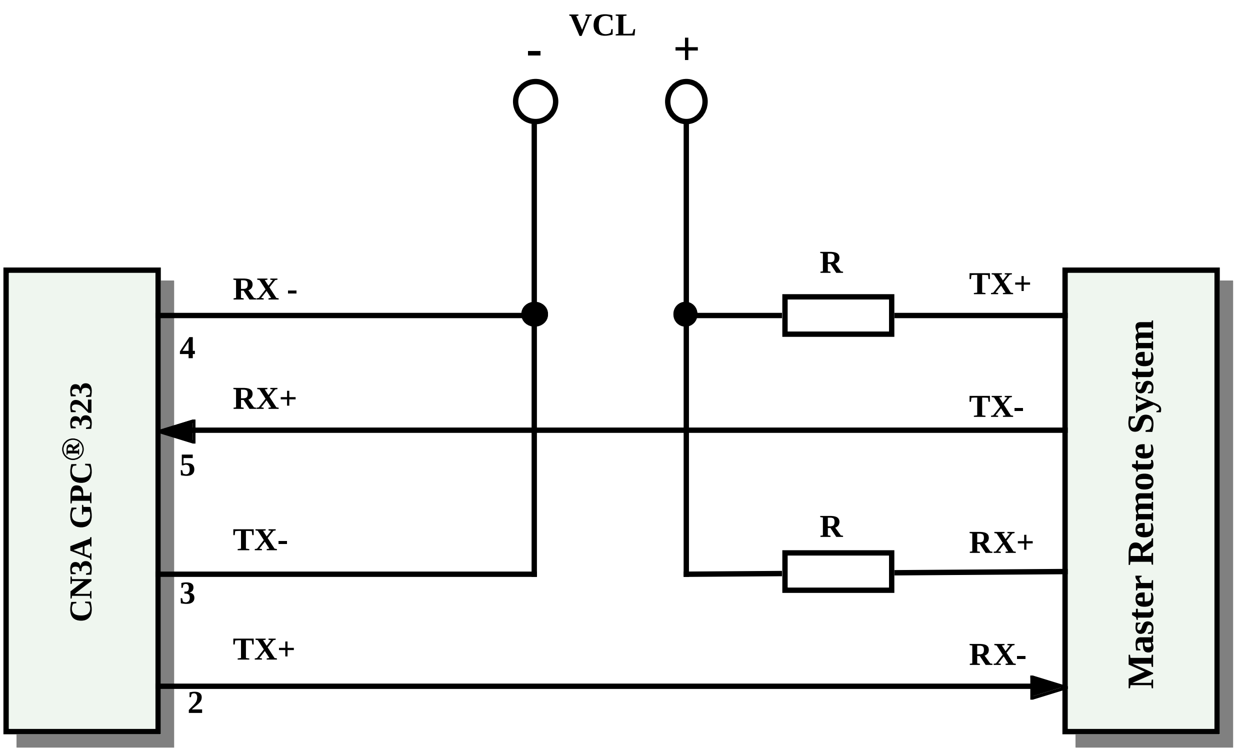


FIGURA 16: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN CURRENT LOOP A 4 FILI

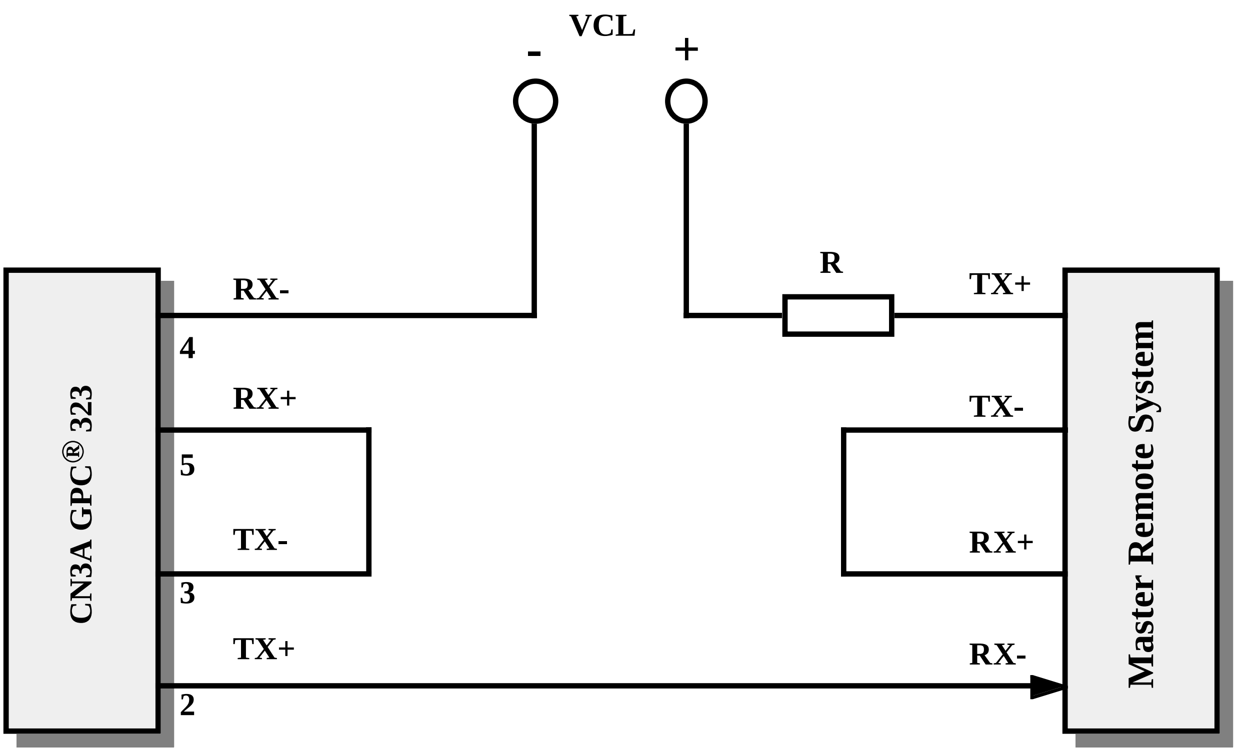


FIGURA 17: ESEMPIO COLLEGAMENTO PUNTO PUNTO IN CURRENT LOOP A 2 FILI

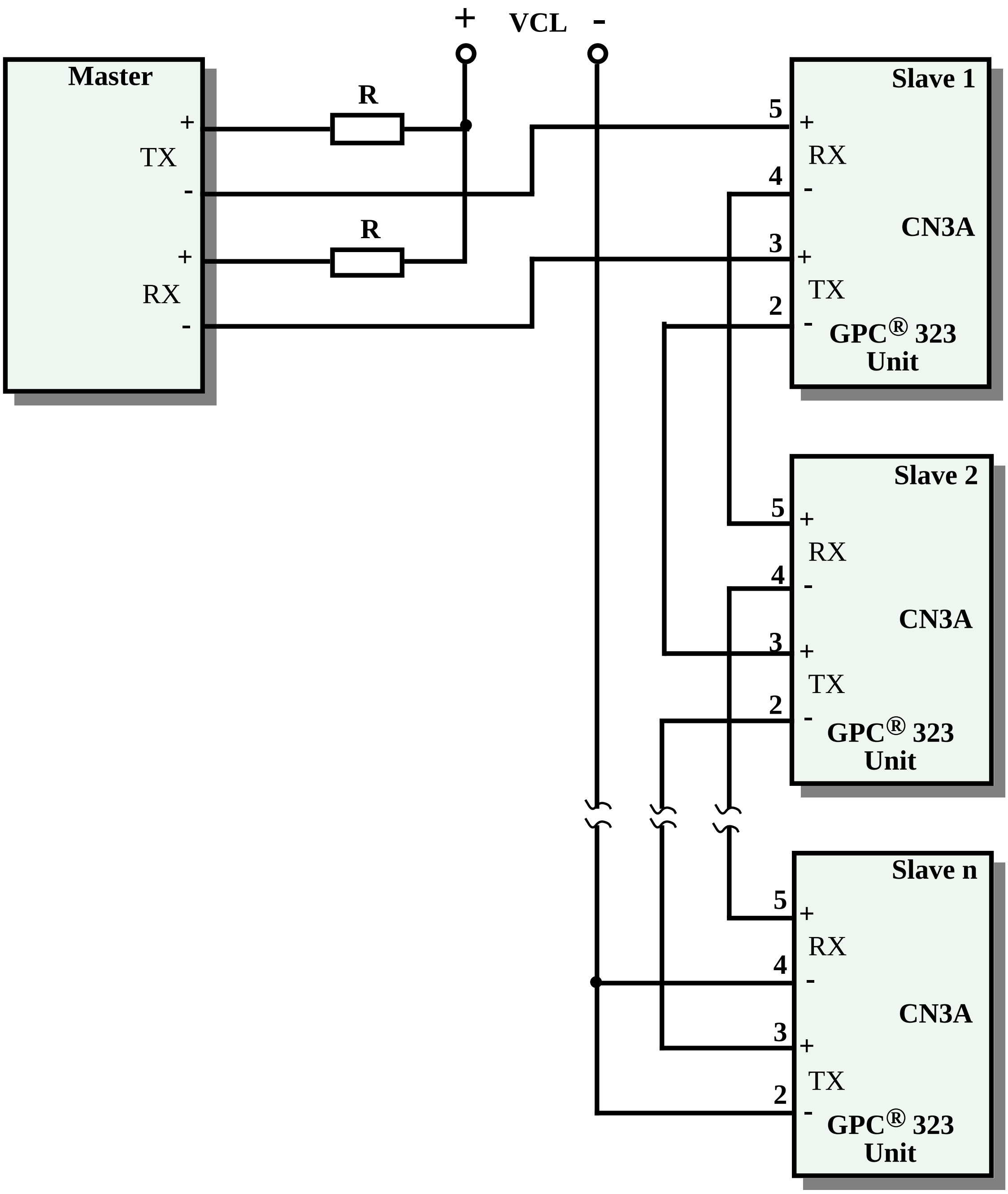


FIGURA 18: ESEMPIO COLLEGAMENTO RETE IN CURRENT LOOP A 4 FILI

Per il collegamento in Current Loop passivo sono possibili due diversi tipi di collegamento: a 2 fili ed a 4 fili. Tali connessioni sono riportate nelle figure 16÷18; in esse é indicata la tensione per alimentare l'anello (**VCL**) e le resistenze di limitazione della corrente (**R**). I valori di tali componenti variano in funzione del numero di dispositivi collegati e della caduta sul cavo di collegamento; bisogna quindi effettuare la scelta considerando che:

- si deve garantire la circolazione di una corrente di **20 mA**;
- su ogni trasmettitore cadono mediamente **2,35 V** con una corrente di 20 mA;
- su ogni ricevitore cadono mediamente **2,52 V** con una corrente di 20 mA;
- in caso di cortocircuito sulla rete ogni trasmettitore dissipi al massimo **125 mW**;
- in caso di cortocircuito sulla rete ogni ricevitore dissipi al massimo **90 mW**.

Per maggiori informazioni consultare il manuale HEWLETT-PACKARD, nella parte che riguarda gli opto accoppiatori per Current Loop denominati **HCPL 4100** e **HCPL 4200**.

CN3B - CONNETTORE PER LINEA SERIALE B

CN3B è un connettore femmina, del tipo PLUG a 6 vie.
Sul connettore sono disponibili i segfnali per la comunicazione della linea seriale B bufferata in RS 232 che é fisicamente collegato alla linea seriale hardware 1 o software del microprocessore. La disposizione di tali segnali, riportata di seguito, è stata studiata in modo da ridurre al minimo le interferenze ed in modo da facilitare la connessione con il campo, mentre i segnali rispettano le normative definite dal CCITT relative ad ognuno degli standard di comunicazione usati.

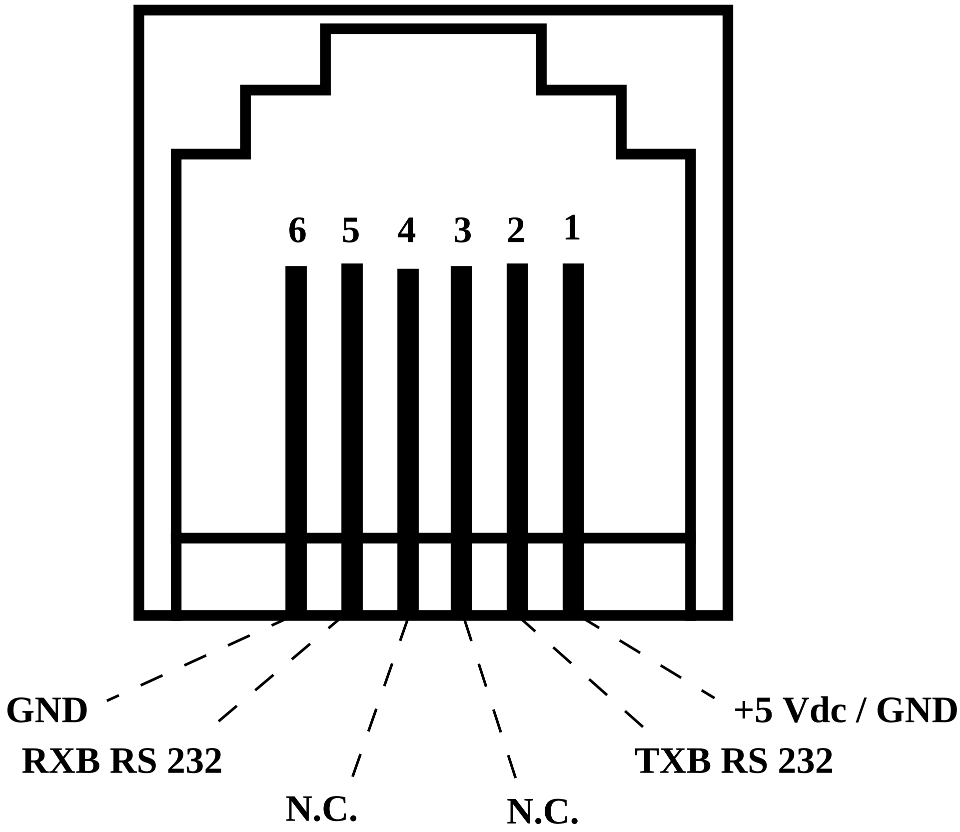


FIGURA 19: CN3B - CONNETTORE PER LINEA SERIALE B

Legenda:

RXB RS232	= I - Receive Data: linea di ricezione in RS 232 della seriale B.
TXB RS232	= O - Transmit Data: linea di trasmissione in RS 232 della seriale B.
+5 Vdc	= O - Linea di alimentazione a +5 Vdc.
GND	= - Linea di massa.
N.C.	= - Non Collegato.

CN2 - CONNETTORE PER BATTERIA ESTERNA DI BACK UP

CN2 é un connettore a scatolino, verticale, maschio, con passo 2,54 mm a 2 vie. Tramite CN2 deve essere collegata una batteria esterna che provvede a mantenere i dati della SRAM di bordo ed a garantire il funzionamento del real time clock, in assenza di tensione di alimentazione (per maggiori informazioni fare riferimento al paragrafo “BACK UP”).

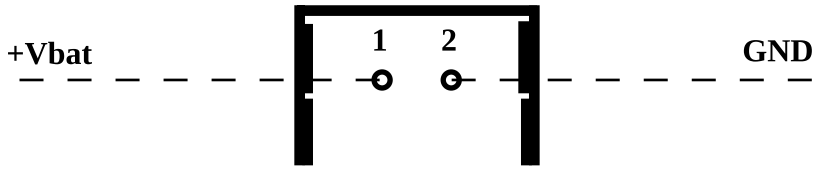


FIGURA 20: CN4 - CONNETTORE PER BATTERIA ESTERNA DI BACK UP

Legenda:

+Vbat	=	I	-	Positivo della batteria esterna di back up
GND	=		-	Negativo della batteria esterna di back up

CN6 - CONNETTORE PER INGRESSI A/D CONVERTER

CN6 è un connettore a scatolino verticale con passo 2.54 mm a 20 piedini. Tramite CN6 si interfacciano le 11 linee di input per la sezione di A/D converter (opzionale) della scheda con il campo esterno. I segnali collegabili a questo connettore sono analogici in tensione (0÷2,49 V o 0÷5,00 V) od in corrente (0÷20, 4÷20 mA) e seguono il pin out standardizzato A/D **ABACO**®. La disposizione dei segnali su questo connettore é studiata in modo da ridurre tutti i problemi di rumore ed interferenza, garantendo quindi un'ottima trasmissione del segnale.

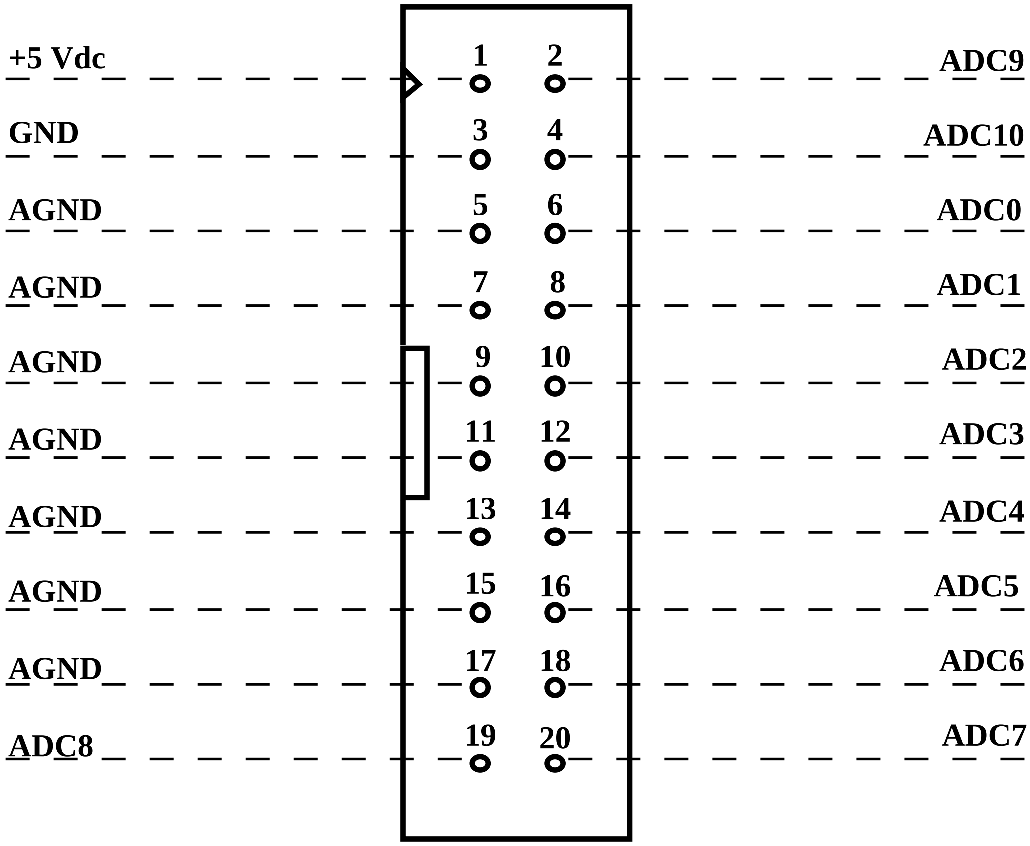


FIGURA 21: CN6 - CONNETTORE PER INGRESSI A/D CONVERTER

Legenda:

ADCn	=	I	- Linea analogica d'ingresso canale n di A/D converter.
GND	=		- Linea di massa.
AGND	=		- Linea di massa analogica.
+5 Vdc	=	O	- Linea di alimentazione a +5 Vdc.

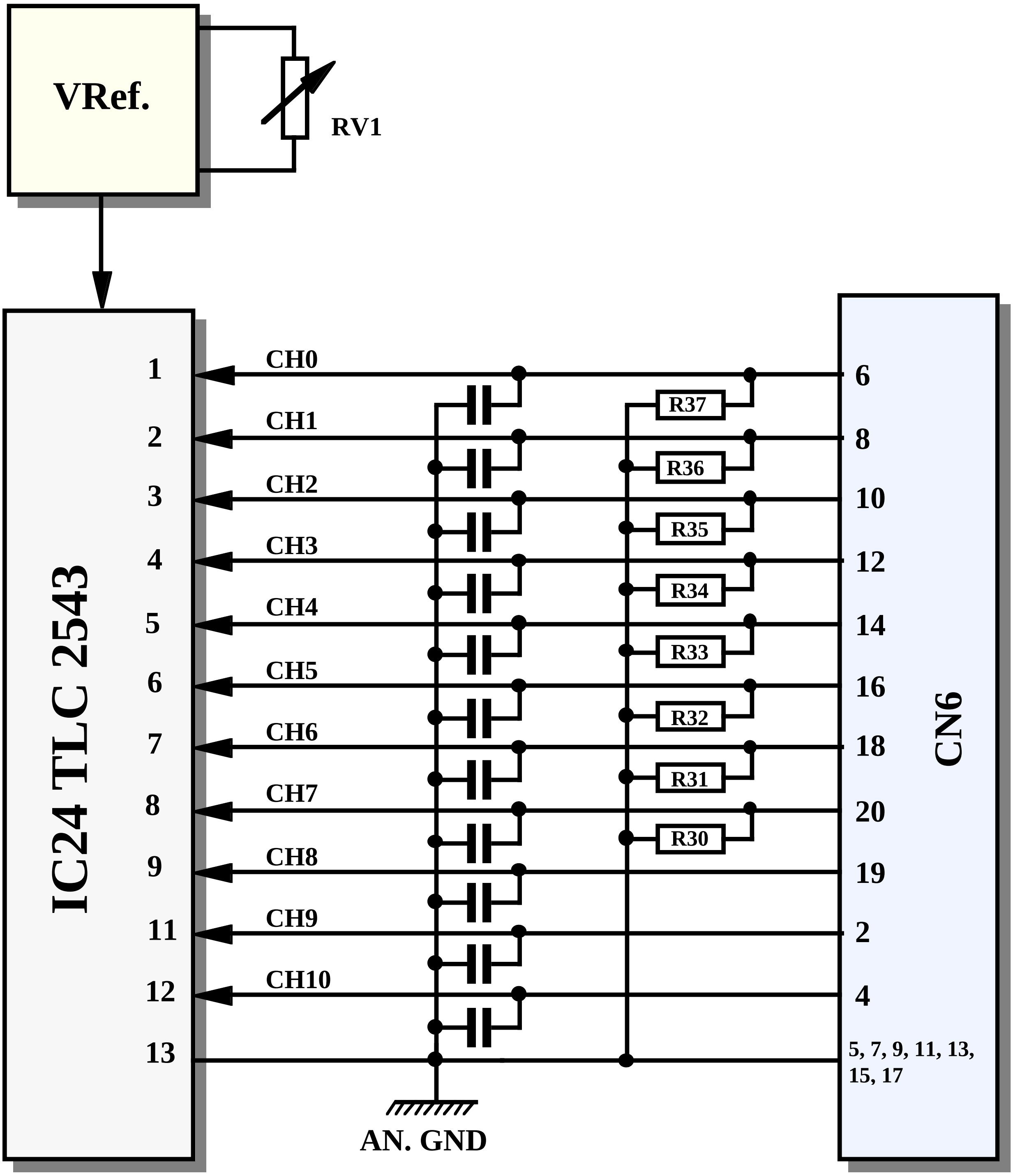


FIGURA 22: SCHEMA D'INGRESSO A/D CONVERTER

INTERFACCE PER I/O DIGITALI

Tramite CN7 (connettore standard di I/O **ABACO®**) si può collegare la **GPC® 323** ai numerosi moduli del carteggio **grifo®** che riportano lo stesso pin out. Dal punto di vista dell'installazione, queste interfacce richiedono solo un flat a 20 vie (codice FLT.20+20) con cui é possibile portare anche le alimentazioni, mentre dal punto di vista software la gestione é altrettanto semplice ed immediata, infatti i pacchetti software disponibili per la **GPC® 323** sono provvisti di tutte le procedure necessarie. Quest'ultime per la maggioranza dei pacchetti software disponibili, coincidono con dei "driver software" aggiunti al linguaggio di programmazione, che consentono di utilizzare direttamente le istruzioni ad alto livello dello stesso linguaggio di programmazione e quindi tutta la loro potenza.
Di particolare interesse é la possibilità di collegare direttamente una serie di moduli come:

- **QTP 24P**, **KDL x24**, **KDF 224**, **DEB 01**, ecc. con cui risolvere tutti i problemi di interfacciamento operatore locale. Questi moduli sono già dotati delle risorse necessarie per gestire un buon livello di colloquio uomo-macchina (includono infatti display alfanumerici, tastiera a matrice e LEDs di visualizzazione) ad una breve distanza dalla **GPC® 323**. Dal punto di vista software i driver disponibili rendono utilizzabili le risorse dell'interfaccia operatore direttamente con le istruzioni ad alto livello per la gestione della console.

- **MCI 64** con cui risolvere tutti i problemi di salvataggio di grosse quantità di dati. Questo modulo é dotato di un connettore per memory card PCMCIA su cui possono essere inserite vari tipi di memory card (RAM, FLASH, ROM, ecc) nei vari size disponibili. Dal punto di vista software i driver disponibili coincidono con un completo file system e rendono utilizzabili le memory card direttamente con le istruzioni ad alto livello per la gestione dei files.

- **IAC 01**, **DEB 01** con cui gestire una stampante con interfaccia parallela CENTRONICS. Quest'ultima può essere collegata direttamente all'interfaccia, con un cavo standard, e quindi gestita con le istruzioni relative alla stampante del linguaggio di programmazione utilizzato.

Per maggiori informazioni relative alle interfaccie per I/O digitali si veda il capitolo "SCHEDE ESTERNE" e la documentazione del software utilizzato.

INTERFACCIAMENTO DEGLI I/O CON IL CAMPO

Al fine di evitare eventuali problemi di collegamento della scheda con tutta l'elettronica del campo a cui la **GPC® 323** si deve interfacciare, si devono seguire le informazioni riportate nei precedenti paragrafi e le relative figure che illustrano le modalità interne di connessione.

- Per i segnali che riguardano la comunicazione seriale con i protocolli RS 232, RS 422, RS 485 o current loop, fare riferimento alle specifiche standard di ognuno di questi protocolli.
- Tutti i segnali a livello TTL possono essere collegati a linee dello stesso tipo riferite alla massa digitale della scheda. Il livello 0V corrisponde allo stato logico 0, mentre il livello 5V corrisponde allo stato logico 1.
- I segnali d'ingresso alla sezione A/D devono essere collegati a segnali analogici a bassa impedenza che rispettino il range di variazione ammesso che può arrivare a +2,49Vdc oppure +5,00 Vdc oppure 0÷20 mA a seconda della configurazione. Per maggiori informazioni si veda il paragrafo sucessivo "SELEZIONE TIPO INGRESSI ANALOGICI".

TASTO DI RESET

Con il tasto P1 presente sulla **GPC® 323** si ha la possibilità di attivare la linea di /RESET della scheda. Una volta premuto il tasto P1, la scheda riprende l'esecuzione del programma in EPROM o FLASH interna, partendo da una condizione di azzeramento generale. La funzione principale di questo tasto é quella di uscire da condizioni di loop infinito, soprattutto durante la fase di debug. Per una facile individuazione di tale pulsante a bordo scheda, si faccia riferimento alla figura 9, mentre per ulteriori informazioni sulla circuiteria di reset si veda il paragrafo "RESET E WATCH DOG".

SELEZIONE TIPO INGRESSI ANALOGICI

La scheda **GPC® 323**, può avere ingressi analogici in tensione e/o corrente, come descritto nei precedenti paragafi e capitoli. La selezione del tipo d'ingresso può essere effettuata solo per otto degli undici canali analogici d'ingresso in fase di ordine della scheda ed é effettuata montando un apposito modulo di conversione corrente-tensione basato su resistenze di caduta di precisione (codice **.8420**). In particolare vale la corrispondenza:

R37	->	canale 0
R36	->	canale 1
R35	->	canale 2
R34	->	canale 3
R33	->	canale 4
R32	->	canale 5
R31	->	canale 6
R30	->	canale 7

Nel caso il modulo corrente-tensione non sia montato (default) il corrispondente canale accetta un ingresso in tensione nei range 0÷2,49 Vdc oppure 0÷5,00 Vdc (opzionale), viceversa un ingresso in corrente.
Il valore della resistenza, su cui si basa il convertitore corrente-tensione, si ottiene dalla seguente formula:

R = 2,49 V / Imax oppure **R = 5,00 V / Imax**

Normalmente i moduti di conversione tensione-corrente si basano su resistenze di precisione da 124Ω, relative ad ingressi 4÷20 mA o 0÷20 mA, per il fondo scala 2,49 Vdc (default).
Per una facile individuazione del modulo e delle resistenze, fare riferimento alle figure 9 e 22.

TEST POINT

La scheda é provvista di un test point chiamato TP1 che permette la lettura della tensione di riferimento dell'A/D converter, regolata in laboratorio:

pin 1	->	Vref
pin 2	->	GND

Per una facile individuazione del test point, si veda la figura 9. Si veda inoltre il prossimo paragrafo.

TRIMMERS E TARATURE

Sulla **GPC® 323** é presente un trimmer da utilizzare per la taratura della scheda. In particolare con il trimmer RV1 si può fissare il valore della tensione di riferimento su cui si basa l'eventuale sezione di A/D converter (codice d'ordine **.AD**). Per una facile individuazione del trimmer a bordo scheda, si faccia riferimento alla figura 9.
La scheda viene sottoposta ad un accurato test di collaudo che provvede a verificare la funzionalità della stessa ed allo stesso tempo a tararla in tutte le sue parti. La taratura viene effettuata in laboratorio a temperatura costante di +20 gradi centigradi, seguendo la procedura di seguito descritta:

- Si effettua la taratura di precisione della Vref della sezione A/D regolando il trimmer RV1, tramite un multimetro galvanicamente isolato a 5 cifre ad un valore di 2,4900 V, misurato sul test point TP1.
- Si verifica la corrispondenza tra segnale analogico fornito in ingresso e combinazione letta dalla sezione A/D converter. La verifica viene effettuata fornendo un segnale di verifica con un calibratore campione e controllando che la differenza tra la combinazione determinata dalla scheda e quella determinata in modo teorico, non superi la somma degli errori della sezione A/D.
- Si blocca il trimmer della scheda, opportunamente tarato, tramite vernice.

Le sezioni d'interfaccia analogica utilizzano componenti di alta precisione che vengono addirittura scelti in fase di montaggio, proprio per evitare lunghe e complicate procedure di taratura. Per questo una volta completato il test di collaudo e quindi la taratura, il trimmer RV1 viene bloccato, in modo da garantire una immunità della taratura anche ad eventuali sollecitazioni meccaniche (vibrazioni, spostamenti, ecc.).
L'utente di norma non deve intervenire sulla taratura della scheda, ma se lo dovesse fare (a causa di derive termiche, derive del tempo, ecc.) deve rigorosamente seguire la procedura sopra illustrata.

RESET E WATCH DOG

La scheda **GPC® 323** è dotata di una circuiteria di watch dog molto efficiente e di facile gestione software. In particolare le caratteristiche di questa circuiteria sono le seguenti:

- funzionamento astabile;
- tempo d'intervento tipico di 1420 msec;
- attivazione via hardware tramite il jumper J8;
- retrigger via software;

Si ricorda che nel funzionamento astabile una volta scaduto il tempo d'intervento la circuiteria si attiva, rimane attiva per il tempo di reset (circa 200 msec) e quindi si disattiva nuovamente. Il jumper J8 collega la circuiteria di watch dog alla circuiteria di reset mentre per l'operazione di retrigger si faccia riferimento al paragrafo WATCH DOG.
Dopo un'attivazione e sucessiva disattivazione del segnale di /RESET, la scheda riprende l'esecuzione del programma salvato su IC5 all'indirizzo 0000H, partendo da una condizione di azzeramento generale di tutti i dispositivi di bordo
Si ricorda inoltre che tra le sorgenti di /RESET della **GPC® 323**, oltre all'eventuale circuiteria di watch dog, sono sempre presenti la circuiteria di power good ed il pulsante P1. Il pulsante P1 è un contatto normalmente aperto e quando il contatto viene chiuso la circuiteria di reset é attivata.

JUMPERS

Esistono a bordo della **GPC® 323** 16 jumpers, di cui 5 a stagno, con cui é possibile effettuare alcune selezioni che riguardano il modo di funzionamento della stessa. Di seguito ne é riportato l'elenco, l'ubicazione e la loro funzione nelle varie modalità di connessione.

JUMPERS	N. VIE	FUNZIONE
J1	2	Collega a + 5 Vdc il pin 26 di CN1.
J2	5	Seleziona il dispositivo di memoria su IC3.
J6	2	Connette la batteria di back up esterna.
J7	3	Seleziona direzionalità e modalità di attivazione della linea seriale in RS 422 e RS 485.
J8	2	Collega la circuiteria di Watch Dog alla circuiteria di reset.
J18	3	Seleziona il tipo di collegamento per il pin 14 (/INT0) della CPU.
J19	3	Fornisce la tensione di programmazione ISP.
J20	3	Seleziona la modalità di partenza della CPU.
J22	2	Collega il segnale TXA RS 232 al pin 2 del connettore CN3A.
J23	2	Collega il segnale RXA RS 232 al pin 5 del connettore CN3A.
J24	3	Collega la linea di ricezione seriale della CPU al driver RS 232 o al driver RS 422-485
JS1, JS2	2	Collegano la circuiteria di terminazione e di forzatura alla linea di comunicazione seriale in RS 422-485.
JS3	3	Seleziona il tipo di collegamento per il pin 1 di CN3A.
JS4	3	Seleziona il tipo di collegamento per il pin 1 di CN3B.
JS6	3	Seleziona quale tensione alimenta la circuiteria che genera la Vref per l'A/D Converter.

FIGURA 23: TABELLA RIASSUNTIVA JUMPERS

Di seguito é riportata una descrizione tabellare delle possibili connessioni dei 16 jumpers con la loro relativa funzione. Per riconoscere tali connessioni sulla scheda si faccia riferimento alla serigrafia della stessa o alla figura 2 di questo manuale, dove viene riportata la numerazione dei pin dei jumpers, che coincide con quella utilizzata nella seguente descrizione. Per l'individuazione dei jumpers a bordo della scheda, si utilizzino invece le figure 26 e 28. In tutte le seguenti tabelle l'***** indica la connessione di default, ovvero quella impostata in fase di collaudo, con cui la scheda viene fornita.

JUMPERS A 5 VIE

JUMPER	CONNESSIONE	UTILIZZO	DEF.
J2	posizione 1-2 e 3-4	Seleziona FLASH EPROM da 32Kbytes.	
	posizione 2-3 e 4-5	Seleziona SRAM o EEPROM da 32Kbytes.	*
	posizione 1-6	Seleziona EPROM da 32Kbytes.	

FIGURA 24: TABELLA JUMPERS A 5 VIE

JUMPERS A 2 VIE

JUMPERS	CONNESSIONE	UTILIZZO	DEF.
J1	non connesso	Non collega il pin 26 di CN1 a +5 Vdc.	
	connesso	Collega il pin 26 di CN1 a +5 Vdc.	*
J6	non connesso	SRAM di IC4 e SRAM+RTC di IC10 tamponati solo dall'eventuale batteria esterna.	*
	connesso	SRAM di IC4 e SRAM+RTC di IC10 tamponati dalla batteria interna e dall'eventuale batteria esterna.	
J8	non connesso	Non collega la circuiteria esterna di Watch Dog alla circuiteria di reset.	*
	connesso	Collega la circuiteria esterna di Watch Dog alla circuiteria di reset.	
J22	non connesso	Non collega la linea di trasmissione della seriale hardware del micro al pin 2 di CN3A.	
	connesso	Collega la linea di trasmissione della seriale hardware del micro al pin 2 di CN3A.	*
J23	non connesso	Non collega la linea di ricezione della seriale hardware del micro al pin 5 di CN3A.	
	connesso	Collega la linea di ricezione della seriale hardware del micro al pin 5 di CN3A.	*
JS1, JS2	non connessi	Non collegano la circuiteria di terminazione e di forzatura alla linea di comunicazione seriale A in RS 422-485.	*
	connessi	Collegano la circuiteria di terminazione e di forzatura alla linea di comunicazione seriale A in RS 422-485.	

FIGURA 25: TABELLA JUMPERS A 2 VIE

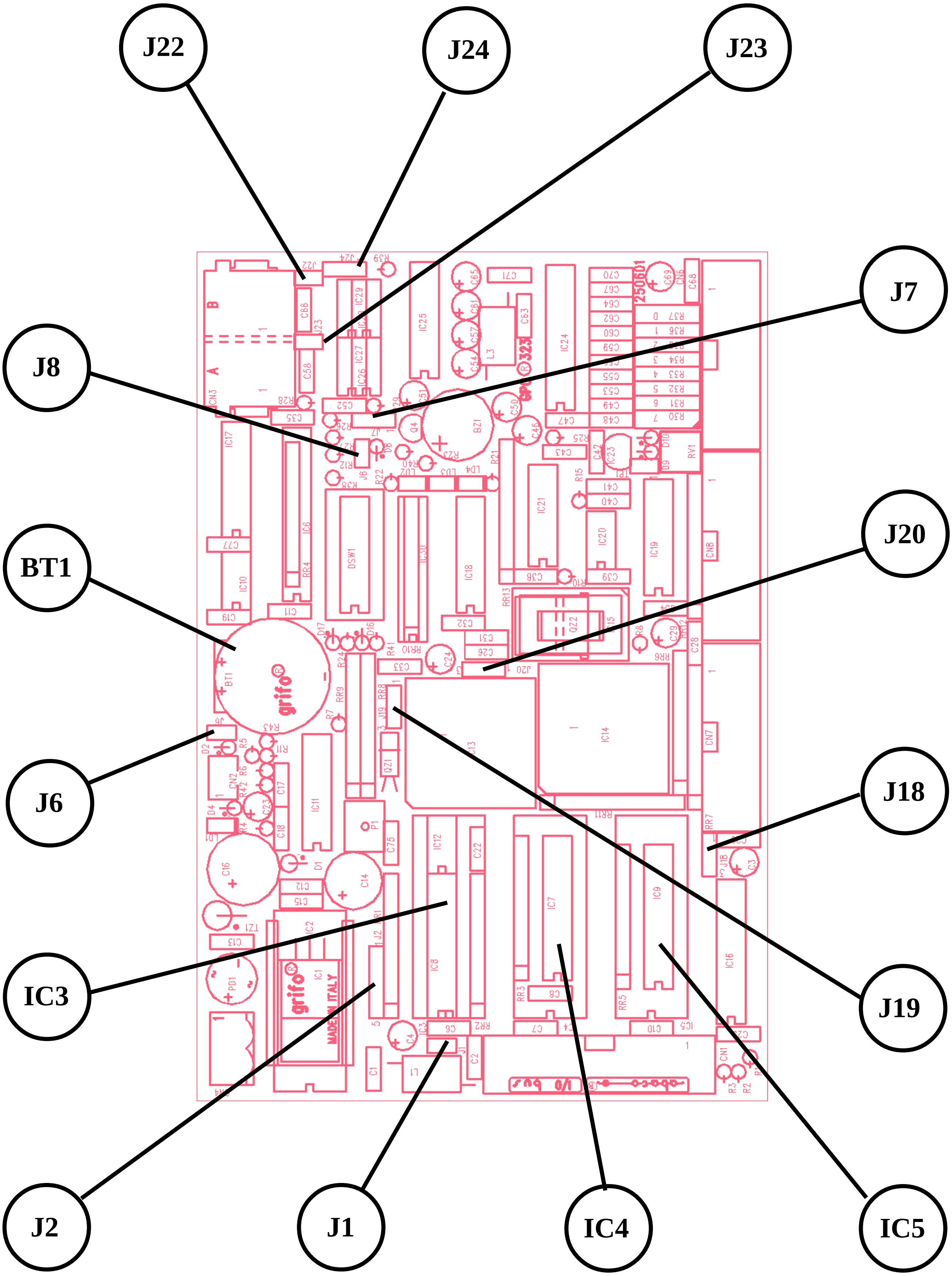


FIGURA 26: DISPOSIZIONE JUMPERS, MEMORIE E BT1 (LATO COMPONENTI)

JUMPERS A 3 VIE

JUMPERS	CONNESSIONE	UTILIZZO	DEF.
J7	posizione 1-2	Collega il segnale di direzione per la comunicazione RS 485.	
	posizione 2-3	Collega il segnale di direzione per la comunicazione RS 422.	*
J18	posizione 1-2	Collega il pin numero 14 del microcontrollore (P3.2, /INT0) al segnale di interrupt /INT dell'ABACO® I/O BUS.	*
	posizione 2-3	Collega il pin numero 14 del microcontrollore (P3.2, /INT0) al segnale di ricezione della linea seriale software.	
J19	posizione 1-2	Collega il pin 35 del microcontrollore (/EA) ai +5 Vdc.	
	posizione 2-3	Collega il pin 35 del microcontrollore (/EA) alla linea di massa.	*
J20	posizione 1-2	Collega il pin 32 del microcontrollore (/PSEN) alla linea di massa.	*
	posizione 2-3	Collega il pin 32 del microcontrollore (/PSEN) alla logica di indirizzamento.	
J24	posizione 1-2	Collega la linea di ricezione della seriale hardware del microcontrollore alla linea di ricezione RS 232.	*
	posizione 2-3	Collega la linea di ricezione della seriale hardware del microcontrollore alla linea di ricezione RS 422-485.	
JS3	posizione 1-2	Collega il pin 1 di CN3A a GND.	*
	posizione 2-3	Collega il pin 1 di CN3A a +5 Vdc.	
JS4	posizione 1-2	Collega il pin 1 di CN3B a GND.	*
	posizione 2-3	Collega il pin 1 di CN3B a +5 Vdc.	
JS6	posizione 1-2	Abilita la Vref di 2,490 Vdc.	*
	posizione 2-3	Abilita la Vref di 5,000 Vdc.	

FIGURA 27: TABELLA JUMPERS A 3 VIE

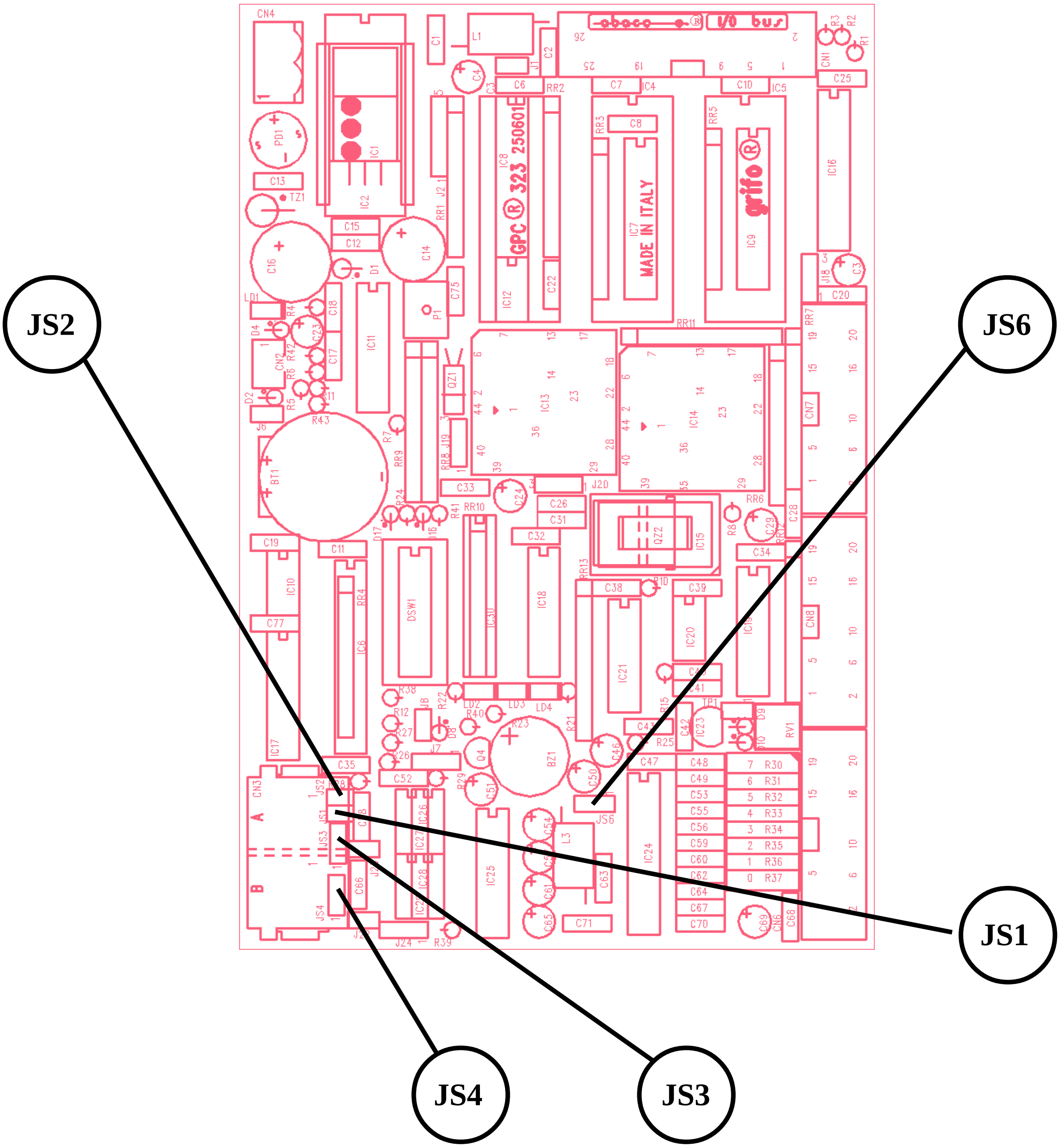


FIGURA 28: DISPOSIZIONE JUMPERS (LATO STAGNATURE)

SEGNALAZIONI VISIVE

La scheda **GPC® 323** é dotata di quattro LEDs con cui segnala alcune condizioni di stato, come descritto nella seguente tabella:

LEDs	COLORE	FUNZIONE
LD1	Rosso	Segnala, quando attivo, la presenza della tensione di alimentazione a +5 Vdc.
LD2	Rosso	Segnala, quando attivo, l'attivazione della circuiteria esterna di Watch Dog.
LD3	Verde	LED di attività temporizzato (spot) gestito via software.
LD4	Verde	LED di attività gestito via software.

FIGURA 29: TABELLA DELLE SEGNALAZIONI VISIVE

La funzione principale di questi LEDs é quella di fornire un'indicazione visiva dello stato della scheda, facilitando quindi le operazioni di verifica di funzionamento di tutto il sistema. Per una più facile individuazione di tali segnalazioni visive, si faccia riferimento alla figura 9.

INPUT DI BORDO

La scheda **GPC® 323** è provvista di 1 dip switch a 8 vie (DSW1), tipicamente utilizzato per la configurazione del sistema, di cui 5 dips sono acquisibili via software dall'utente (DIP 1÷4,8), due sono usati per la selezione della configurazione memorie (DIP 5,6) ed uno non é utilizzato (DIP 7). Le applicazioni più immediate possono essere quelle destinate al settaggio delle condizioni di lavoro od alla selezione di parametri relativi al firmware di bordo. La lettura della combinazione fissata sul dip switch avviene in logica negata (0 -> dip in ON ed 1 -> dip in OFF) effettuando un'operazione di lettura all' indirizzo dedicatogli dalla logica di controllo della scheda. Per ulteriori informazioni si faccia riferimento ai paragrafi "MAPPAGGIO DELL'I/O" e "MAPPAGGIO DELLE MEMORIE", mentre per una facile individuazione della sua posizione si veda la figura 9.

BACK UP

La **GPC® 323** é provvista di una batteria al litio BT1 che provvede a tamponare la SRAM e l'RTC di bordo anche in assenza della tensione di alimentazione. Il jumper J6 provvede a collegare o meno questa batteria in modo da salvaguardarne la durata prima dell'installazione o in tutti i casi in cui il back up non é necessario. Una seconda batteria esterna può essere collegata alla circuiteria di back up tramite il connettore CN2: quest'ultima non é interessata dalla configurazione del jumper J6 e sostituisce a tutti gli effetti la BT1. L'utente può ordinare una batteria esterna (2,1 Ah) pronta per essere collegata a CN2 con il codice: **.LITIO**.
Per la scelta della batteria esterna di back up seguire le indicazioni del paragrafo "CARATTERISTICHE ELETTRICHE", mentre per la sua individuazione si veda la figura 9.

INTERRUPTS

Una caratteristica peculiare della **GPC® 323** è la notevole potenza nella gestione delle interruzioni. Di seguito viene riportata una breve descrizione di come possono essere gestiti i segnali hardware di interrupt della scheda; per quanto riguarda la gestione di tali interrupts si faccia riferimento ai data sheets del microprocessore oppure all'appendice B di questo manuale.

-Periferiche della CPU:	Le possibili sorgenti d'interrupt interno sono le sezioni: timer counter 0÷2; linee seriali 0, 1; interrupt esterni 0÷5; watch dog interno, ecc.
- Real Time Clock:	E' collegato in open collector al pin /INT1 = P3.3 della CPU.
- Power failure:	E' collegato in open collector al pin /INT1 = P3.3 della CPU.
- Seriale software:	E' collegato in open collector al pin /INT0 = P3.2 della CPU, a seconda dello stato di J24.
- **ABACO®** I/O BUS:	/INT BUS di CN1 é collegata al pin /INT0 = P3.2 della CPU, a seconda dello stato di J24. /NMI BUS di CN1 é direttamente collegata al pin T2 = P1.0 della CPU.

L'ultimo collegamento é particolarmente importante per due diversi motivi: ogni attivazione del segnale /NMI BUS può generare un interrupt oppure ogni variazione dello stesso segnale può essere contata. La gestione del segnale /NMI BUS é definita con la programmazione del timer counter 2, quindi l'utente può selezionare il modo preferito via software. Questa caratteristica é veramente interessante se utilizzata in abbinamento alle schede **ZBT xxx** e **ZBR xxx**. Con quest'ultime infatti dei segnali digitali optoisolati d'ingresso possono generare interrupts o essere semplicemente contati.
Il microprocessore gestisce una catena di priorità hardware che regolamenta l'attivazione contemporanea di più interrupts. Gli indirizzi delle procedure di risposta agli interrupts possono essere programmati arbitrariamente dall'utente nell'area codice, mentre i livelli di priorità e le attivazioni sono definibili tramite i registri interni al microprocessore. In questo modo l'utente ha sempre la possibilità di rispondere in maniera efficace e veloce a qualsiasi evento esterno, stabilendo anche la priorità delle varie sorgenti.

POWER FAILURE

In abbinamento alla circuiteria di power management gestita dalla CPU della **GPC® 323** é inoltre disponibile un'interessante circuiteria di power failure. Quest'ultima è collegata all'interrupt /INT0 del microprocessore.
La circuiteria si preoccupa di controllare la tensione di alimentazione e quando questa scende al valore di soglia (**52 mV** prima dell'intervento del reset), provvede ad attivare l'uscita richiedendo l'attenzione della CPU. Da notare che il tempo che intercorre tra l'attivazione del power failure e quello del reset, varia in funzione del tipo di alimentazione della scheda; questo normalmente é nell'ordine dei 100 µsec, sufficienti solo per eseguire procedure di risposta veloci (ad esempio il salvataggio di un flag nella memoria tamponata).
L'uso classico della circuiteria di power failure é quello di informare la scheda dell'imminente caduta della tensione di alimentazione, in modo da salvare le necessarie condizioni di stato, grazie alla carica residua della stessa sezione alimentatrice.

SELEZIONE MEMORIE

La **GPC® 323** può montare fino ad un massimo di 98K bytes di memoria variamente suddivisa. In particolare valgono le informazioni riportate nella seguente tabella:

IC	DISPOSITIVO	DIMENSIONE	STRIPPAGGIO
3	SRAM/EEPROM	32K Bytes	J2 in 2-3 e 4-5
	EPROM	32K Bytes	J2 in 3-4
	FLASH EPROM	32K Bytes	J2 in 1-2 e 3-4
4	SRAM/EEPROM	32K Bytes	-
5	EPROM	32K Bytes	-
10	EEPROM seriale	512÷2048 Bytes	-
12	SRAM+RTC	256 Bytes	-

FIGURA 30: TABELLA DI SELEZIONE MEMORIE

Gli zoccoli IC3, IC4 e IC5 seguono lo standard JEDEC, quindi i dispositivi di memoria che possono essere installati dovranno essere conformi a questo standard. Il settaggio dei jumpers indicato nella figura 21 influisce solo sulla configurazione degli zoccoli indicati nella tabella stessa, mentre gli indirizzi di mappaggio delle memorie sono definiti da altri jumers, come descritto nel paragrafo “MAPPAGGIO DELLE MEMORIE”. Per vedere la posizione delle memorie si faccia riferimento alla figura 26.
Normalmente la **GPC® 323** viene fornita nella configurazione di default che comprende 32K SRAM installati su IC4 e 512 bytes di EEPROM installati su IC10; ogni configurazione diversa da questa deve essere specificata in fase di ordine o montata autonomamente dall'utente. Di seguito vengono riportati i codici delle opzioni di memoria disponibili:

.32K	->	32K x 8 SRAM
.32KMOD	->	32K x 8 SRAM tamponata
.32EE	->	32K x 8 EEPROM parallela
.32KF	->	32K x 8 FLASH EPROM parallela
.EE02	->	2K bit (256 byte) EEPROM seriale
.EE08	->	8K bit (1K byte) EEPROM seriale
.EE16	->	16K bit (2K byte) EEPROM seriale

Per ulteriori informazioni si prega di contattare direttamente la **grifo®**.

JUMPER A STAGNO

La connessione di default dei jumpers a stagno denominati **JSxx**, é effettuata con una sottile pista sul lato stagnature. Quindi, se tale configurazione deve essere variata, si deve prima tagliare la pista con un taglierino affilato e poi effettuare la connessione richiesta con uno stagnatore di bassa potenza utilizzando dello stagno non corrosivo.

TENSIONI DI ALIMENTAZIONE

La scheda **GPC® 323** dispone di una efficiente circuiteria che si presta a risolvere in modo comodo ed efficace il problema dell'alimentazione della scheda in qualsiasi condizione di utilizzo. Di seguito vengono riportate le possibili configurazioni della sezione alimentatrice:

- *Senza sezione alimentatrice* *(default):*
  In questa configurazione la scheda deve essere alimentata da una tensione di +5 Vdc che può essere fornita sugli appositi pin di CN1 (**+5 Vdc** su pin 26, **GND** su pin 25).
- *Con sezione alimentatrice lineare* *(opzione **.ALIM12**):*
  In questa configurazione la scheda deve essere alimentata da una tensione di 6÷12 Vac, o corrispondente tensione continua, che deve essere fornita sui pin 1 e 2 di CN4.
- *Con sezione alimentatrice switching* *(opzione **.SW**):*
  In questa configurazione la scheda deve essere alimentata da una tensione di 8÷24 Vac, o la corrispondente tensione continua (12÷34 Vdc), che deve essere fornita sui pin 1 e 2 di CN4.

Indipendentemente dalla sezione alimentatrice scelta la **GPC® 323** é sempre dotata di un'efficace circuiteria di protezione che si preoccupa di proteggere la scheda da sovratensioni o dal rumore del campo. Si ricorda che la selezione del tipo di sezione alimentatrice della scheda deve avvenire in fase di ordine della stessa; infatti questa scelta implica una diversa configurazione hardware, che deve essere effettuata dal personale addetto.
Il jumper J1 collega il positivo della tensione di alimentazione della scheda al connettore dell'**ABACO®** I/O BUS e deve quindi essere non connesso solo quando si collega la scheda provvista di sezione alimentatrice ad un sistema a sua volta provvisto di propria sezione alimentatrice.
Al fine di ridurre i consumi della scheda si possono utilizzare le modalità operative di IDLE e STOP MODE della CPU. Per ulteriori informazioni su queste modalità si prega di consultare l'appendice B di questo manuale o la documentazione dei produttori delle CPU.
Per ulteriori informazioni si faccia riferimento al paragrafo "CARATTERISTICHE ELETTRICHE".

PROGRAMMAZIONE IN SYSTEM (ISP)

Una delle caratteristiche più importanti della **GPC® 323** é la possibilità di usare i nuovi microprocessori della PHILIPS 89CRx+/2 che supportano la programmazione in system, ovvero la programmazione effettuata bordo scheda, senza dover togliere la CPU. Di seguito sono descritti i passi da effettuare:

1) sviluppare il programma applicativo tramite un pacchetto software che generi un codice eseguibile
2) connettere il jumper J19 in posizione 1-2 e J20 in posizione 1-2
3) collegare la linea seriale A in RS 232 ad una linea COM libera di un personal computer
4) alimentare la scheda
5) programmare la FLASH EPROM interna del microprocessore usando l'apposito programma fornito dalla PHILIPS: **WINISP**.
6) togliere alimentazione alla scheda
7) ricollegare J19 e J20 in posizione 2-3
8) rialimentare la scheda: il programma applicativo é eseguito dalla FLASH ROM interna.

L'ISP riduce i costi complessivi dell'applicazione, infatti elimina l'uso di EPROM, programmatore di EPROM, FLASH EPROM, ecc. Per ulteriori informazioni relative alla programmazione ISP fare riferimento alla specifica documentazione tecnica della PHILIPS.

COMUNICAZIONE SERIALE

La linea di comunicazione seriale A della scheda **GPC® 323** può essere bufferata in RS 232, RS 422, RS 485 o in current loop. La selezione del tipo d'interfacciamento avviene via hardware e viene effettuata tramite un opportuno strippaggio dei jumpers di bordo, come può essere desunto dalla lettura delle precedenti tabelle. Dal punto di vista software sono invece definibili tutti i parametri del protocollo fisico di comunicazione (baud rate, stop bit, bit per carattere, ecc) tramite la programmazione dei registri interni della CPU.
Alcuni componenti necessari per le configurazioni RS 422, RS 485 e current loop non sono montati e collaudati sulla scheda in configurazione di default; per questo la prima configurazione della seriale A non in RS 232 deve essere sempre effettuata dai tecnici **grifo®**. A questo punto l'utente può cambiare autonomamente la configurazione seguendo le informazioni sotto riportate (da notare che i jumpers non menzionati nella sucessiva descrizione non hanno alcuna influenza ai fini della comunicazione, qualunque posizione essi occupino):

- LINEA SERIALE A SETTATA IN RS 232 (configurazione default)

			IC25	= driver MAX 202
J7	=	indifferente	IC26	= nessun componente
J22, J23	=	connessi	IC27	= nessun componente
J24	=	posizione 2-3	IC28	= nessun componente
JS1, JS2	=	non connessi	IC29	= nessun componente

- LINEA SERIALE A SETTATA IN CURRENT LOOP (opzione .CLOOP)

			IC25	= nessun componente
J7	=	indifferente	IC26	= driver HCPL 4100
J22, J23	=	non connessi	IC27	= nessun componente
J24	=	posizione 2-3	IC28	= driver HCPL 4200
JS1, JS2	=	non connessi	IC29	= nessun componente

Da ricordare che l'interfaccia seriale in current loop é di tipo passivo e si deve quindi collegare una linea current loop attiva, ovvero provvista di un proprio alimentatore. L'interfaccia current loop può essere utilizzata per realizzare sia connessioni punto punto che multipunto con un collegamento a 4 o 2 fili.

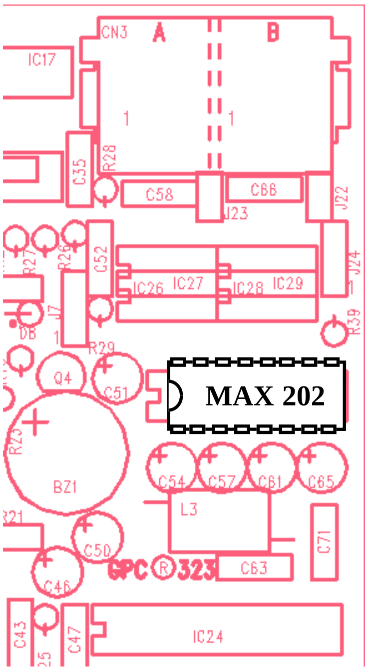
- LINEA SERIALE A SETTATA IN RS 422 (opzione .RS 422)

			IC25	= nessun componente
J7	=	posizione1-2	IC26	= nessun componente
J22, J23	=	non connessi	IC27	= driver SN75176 o MAX 483
J24	=	posizione 2-3	IC28	= nessun componente
JS1, JS2	=	(*)	IC29	= driver SN75176 o MAX 483

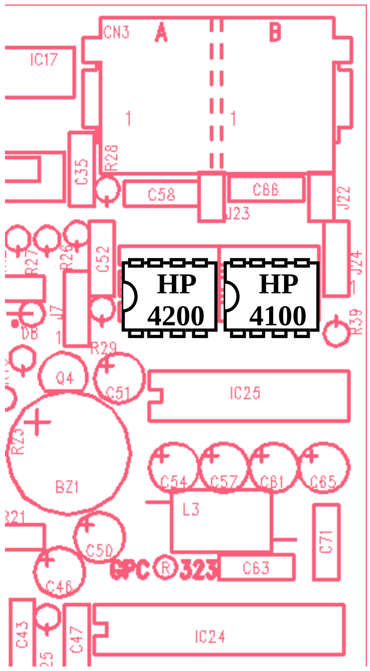
Lo stato del segnale DIR, gestito via software, consente di abilitare o disabilitare il trasmettitore come segue:

DIR = livello basso	= stato logico 0	->	trasmettitore attivo
DIR = livello alto	= stato logico 1	->	trasmettitore disattivo

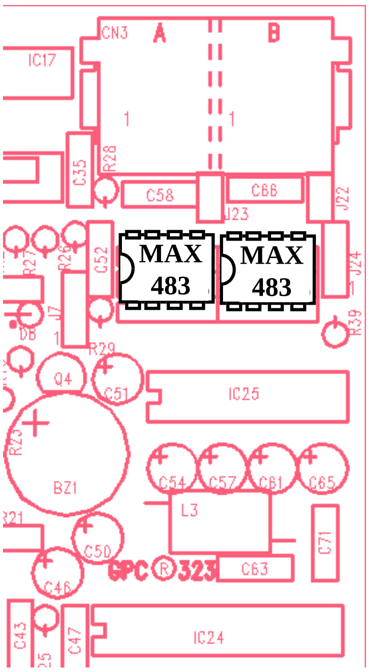
Per sistemi punto punto, la linea DIR può essere mantenuta sempre bassa (trasmettitore sempre attivo), mentre per sistemi multipunto si deve attivare il trasmettitore solo in corrispondenza della trasmissione.



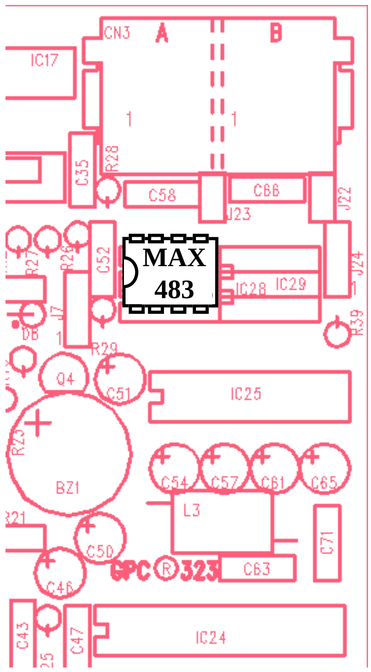
Seriale A in RS 232



Seriale A in current loop



Seriale A in RS 422



Seriale A in RS 485

FIGURA 31: DISPOSIZIONE DRIVER PER COMUNICAZIONE SERIALE

- LINEA SERIALE A SETTATA IN RS 485 (opzione .RS 485)

			IC25	= nessun componente
J7	=	posizione 2-3	IC26	= nessun componente
J22, J23	=	non connessi	IC27	= driver SN75176 o MAX 483
J24	=	posizione 2-3	IC28	= nessun componente
JS1, JS2	=	(*)	IC29	= nessun componente

In questa modalità le linee da utilizzare sono i pin 4 e 5 di CN3A, che quindi diventano le linee di trasmissione o ricezione a seconda dello stato del segnale DIR, gestito via software, come segue:

DIR = livello basso = stato logico 0 -> linea in trasmissione
DIR = livello alto = stato logico 1 -> linea in ricezione

Questa comunicazione la si utilizza sia per connessioni punto punto che multipunto con un collegamento a 2 fili. Sempre in questa modalità si riceve quanto trasmesso, in modo da fornire al sistema la possibilità di verificare autonomamente la riuscita della trasmissione; infatti in caso di conflitti sulla linea, quanto trasmesso non viene ricevuto correttamente e viceversa.

(*) Nel caso si utilizzi la linea seriale in RS 422 o RS 485, con i jumpers JS1 e JS2 é possibile connettere la circuiteria di terminazione e forzatura sulla linea . Tale circuiteria deve essere sempre presente nel caso di sistemi punto punto, mentre nel caso di sistemi multipunto, deve essere collegata solo sulle schede che risultano essere alla maggior distanza, ovvero ai capi della linea di comunicazione.

In fase di reset o power on, il segnale DIR è mantenuto a livello logico alto di conseguenza in seguito ad una di queste fasi il driver RS 485 è in ricezione o il driver di trasmissione RS 422 è disattivo, in modo da eliminare eventuali conflittualità sulla linea di comunicazione.

La **GPC® 323** dispone di una seconda linea di comunicazione seriale (B) che può essere bufferata solo in RS 232. Tale linea ha caratteristiche diverse a seconda del microcontrollore usato, come di seguito descritto:

- µP 80C32 e compatibili

La linea seriale B è una linea seriale software gestita tramite due linee di I/O del microcontrollore (pin 4 = P1.2 = RXB e pin 5 = P1.3 = TXB). I parametri della comunicazione sono quindi definibili via software parametrizzando il firmware di gestione (per maggiori informazioni fare riferimento al manuale d'uso del pacchetto software).

- µP 80C320 e compatibili

La linea seriale B è una linea seriale hardware gestita direttamente dalla seriale 1 del microcontrollore (pin 4 = RXB = RX2 e pin 5 = TXB = TX2). I parametri della comunicazione sono quindi definibili via software accedendo agli appositi registri interni (consultarela documentazione della casa costruttrice o l'appendice B di questo manuale).

Per ulteriori informazioni relative alla comunicazione seriale fare riferimento agli esempi di collegamento delle figure 12÷18 ed al paragrafo DIREZIONALITA' RS 422, RS 485.

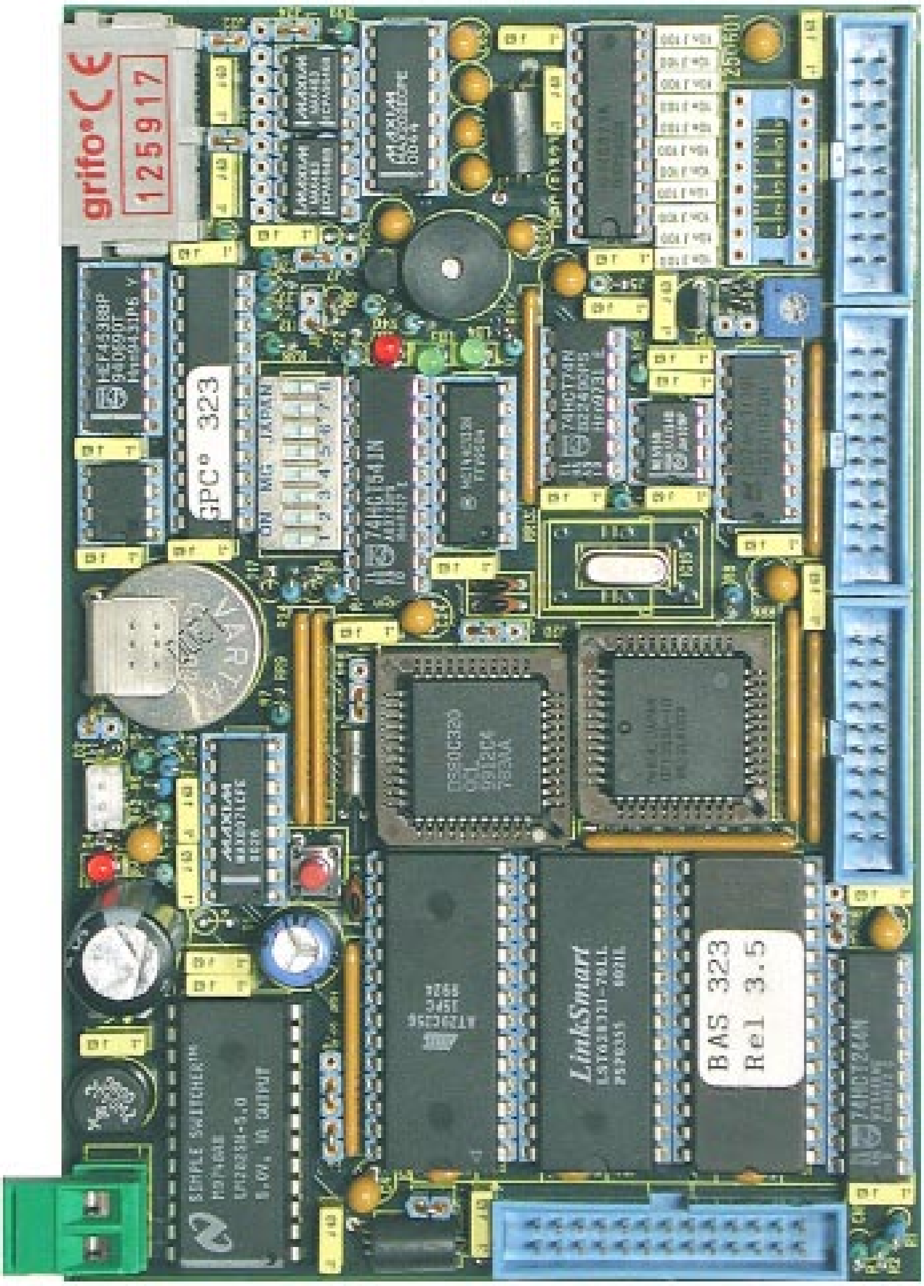


FIGURA 32: VISTA DELLA SCHEDA

DESCRIZIONE SOFTWARE

Questa scheda ha la possibilità di usufruire di una ricca serie di strutture software che consentono di utilizzarne al meglio le caratteristiche. In generale la scheda può sfruttare tutte le risorse software per il microprocessore montato e tutti i pacchetti ideati per la famiglia 51, sia ad alto che a basso livello. Tra questi ricordiamo:

GET51: Completo programma di EDITOR , Comunicazione e gestione delle Memorie di Massa per le schede della famiglia 51. Questo programma, sviluppato dalla **grifo®**, consente di operare in condizioni ottimali, in abbinamento ai pacchetti software BASIC 324, MDP, BXC51, FMO52, ecc. Una serie di comodi menù a tendina facilita l'uso del programma, il quale può funzionare anche in abbinamento ad un mouse. Il programma, oltre che girare in ambiente MS-DOS, gira tranquillamente anche sulle macchine MACINTOSH in abbinamento al programma VIRTUAL-PC. Viene fornito su dischetti MS-DOS da 3"1/2.

MDP: monitor debugger in grado di caricare e debuggare un qualsiasi file HEX con codice'I51. Dispone di tutti i comandi normalmente disponibili con un'emulatore e fornisce quindi all'utente la possibilità di operare comodamente con tutte le risorse di bordo. Per questo pacchetto software é sufficiente disporre di un P.C. che effettua le sole operazioni di console nei confronti dell'utente.

FMO52: monitor debugger in grado di caricare e debuggare un qualsiasi file HEX con codice'I51. Dispone di tutti i comandi normalmente disponibili con un'emulatore e fornisce quindi all'utente la possibilità di operare comodamente con tutte le risorse di bordo. Per questo pacchetto software é sufficiente disporre di un P.C. che effettua le sole operazioni di console nei confronti dell'utente.
E' inoltre in grado di programmare su FLASH EPROM l'applicativo sviluppato dall'utente e sucessivamente eseguirlo in modalità di autorun.

FORTH: completa struttura di sviluppo che consente di programmare la scheda in FORTH. Richiede un P.C. per l'interfaccia utente e rende disponibili strutture dati e di programmazione ad alto livello, che velocizzano lo sviluppo dell'applicativo con ottime caratteristiche in termini di codice sviluppato e velocità di esecuzione.

BASIC 323: completa struttura di sviluppo che consente di programmare la scheda con un BASIC interpretato adatto alle applicazioni industriali. Per opearare é sufficiente un P.C. che svolge le funzioni di consolle nei confronti della scheda su cui viene invece sviluppato, debuggato, provato e salvato il programma da realizzare. La programmazione é ad alto livello ed interessa la maggioranza dei dispositivi a bordo scheda, di cui vengono già forniti i driver software di facile utilizzo.

BXC51: Cross compilatore per files sorgenti scritti in BASIC 324. Disponibile in ambiente MS-DOS, permette un notevole incremento in termini di velocità di esecuzione rispetto all'equivalente programma in BASIC interpretato.

MCA 51: Macro Cross Assembler. Disponibile in ambiente MS-DOS e nella versione assoluta o rilocabile, permette una facile ed efficiente programmazione in assembler, dei microcontrollori basati sull'8051. In versione rilocabile, viene anche fornito un linker ed un gestore di librerie.

MCC 51: Integer Cross Compiler per files sorgenti scritti in linguaggio C. Disponibile in ambiente MS-DOS, genera un source assembly compatibile con il MICRO/ASM 51 o con il macro assembler rilocabile dell'Intel (MCS-51).

MCS 51: Simulatore e Debugger a livello source. Simulatore/Debugger in grado di simulare i microcontrolloridella famiglia I51 e di monitorare lo stato di esecuzione di un programma. Permette tramite un PC e senza l'aggiunta di emulatori o hardware addizionale, il caricamento o il salvataggio di file HEX o simbolici, il settaggio di breakpoints, l'esecuzione in modo trace di istruzioni C e/o assembler, la visualizzazione di qualsiasi registro o variabile, ecc.

MCK 51: E' la somma dei pacchetti MCC 51 e MCA 51 e coincide con un completo compilatore C in grado di generare codice eseguibile per la famiglia '51 Intel e di generare un file simbolico utilizzabile dall'MCS 51.

HI TECH C 51: Cross compilatore per file sorgenti scritti in linguaggio C. E' un potente pacchetto software che tramite un comodo I.D.E. permette di utilizzare un editor, un compilatore C (floating point), un assemblatore, un ottimizzatore, un linker e un remote debugger. Sono inoltre inclusi i source delle librerie.

SYS51CW: Cross compilatore per programmi scritti in C, disponibile in ambiente WINDOWS con un comodo IDE che mette a disposizione: editor, compilatore C, assemblatore, ottimizzatore, linker, librerie ed un debugger simbolico remoto.

SYS51PW: Cross compilatore per programmi scritti in PASCAL, disponibile in ambiente WINDOWS con un comodo IDE che mette a disposizione: editor, compilatore PASCAL, assemblatore, ottimizzatore, linker, librerie ed un debugger simbolico remoto.

DDS MICRO C 51: E' un comodo pacchetto software, a basso costo, che tramite un completo I.D.E. permette di utilizzare un editor, un compilatore C (integer), un assemblatore, un linker e un remote debugger abbinato ad un monitor. Sono inclusi i sorgenti delle librerie ed una serie di utility.

NOICE 51: Potente struttura di debugger composta da un monitor debugger residente sulla scheda e da un apposito programma MS-DOS. I due programmi comunicano tramite una linea seriale in RS 232. Il NOICE include: debug a livello sorgente, disassemblatore, visualizzatore di file, editor e visualizzazione della memoria, numero di breakpoint illimitato, esecuzione di singole istruzioni indipendente dall'hardware, definizione di simboli, possibilità di eseguire file di comandi, gestione del back trace, help in linea, ecc.

OPEN 51/UNI: Emulatore in circuit per la famiglia '51 Intel. E' un potente pacchetto hardware e software che include: debug a livello sorgente e simbolico, gestione di progetti, editor multi finestra, esecuzione di compilatori, assemblatori esterni, debug di più moduli contemporaneo, disassemblatore, funzioni di step e trace a livello sorgente, funzioni di animazione, veloce gestione dei breakpoint sempre a livello sorgente, visualizzazione e modifica di variabili, anche strutturate, ad alto livello.

BASCOM 8051: Cross compilatore a basso costo per files sorgenti scritti in BASIC, disponibile in ambiente WINDOWS con un comodo IDE che mette a disposizione un editor, il compilatore ed un simulatore molto potente per il debugger del sorgente. Comprende molti modelli di memoria, svariati tipi di dati ed istruzioni dedicate alle risorse hardware.

SOFTICE: E' un debugger simbolico remoto con cross assemblatore. Dispone dei classici comandi disponibili su un emulatore hardware ma richiede solo un P.C. collegato tramite una linea seriale. Una interfaccia utente ad alto livello, provvista di numerose finestre separate, mostra lo stato del processore e della scheda.

INTRODUZIONE

In questo capitolo ci occuperemo di fornire tutte le informazioni relative all'utilizzo della scheda, dal punto di vista della programmazione via software. Tra queste si trovano le informazioni riguardanti il mappaggio della scheda e la gestione software delle sezioni componenti.

MAPPAGGIO DELLE RISORSE DI BORDO

La gestione delle risorse della scheda è affidata ad una logica di controllo completamente realizzata con logiche programmabili. Essa si occupa del mappaggio delle memorie e di tutte le periferiche di bordo, semplificando l'operatività dell' utente. La logica di controllo è realizzata in modo da gestire separatamente il mappaggio delle memorie di bordo ed il mappaggio delle periferiche viste come input/output. Complessivamente le CPU disponibili indirizzano direttamente 64K di area codice e 64K di area dati, quindi alla logica di controllo è assegnato il compito di allocare i dispositivi di memoria installabili nello spazio fisico massimo di 128K Bytes. Questa gestione è effettuata via hardware tramite lo strippaggio di alcuni jumpers (J2) con cui si può definire quali memorie utilizzare e il range di indirizzamento per ciascuna di esse. Per quanto riguarda il mappaggio dell'I/O si deve invece ricordare che la logica di controllo provvede naturalmente a non utilizzare le locazioni riservate per le periferiche interne della CPU, in modo da evitare ogni problema di conflittualità. Riassumendo i dispositivi mappati sulla scheda sono:

- 32K bytes di EPROM su IC 5
- 32K bytes di SRAM su IC 4
- 32K bytes di SRAM, EPROM, EEPROM, FLASH EPROM su IC 3
- **ABACO® I/O BUS**
- Dip Switch DSW1
- LEDs utente
- Buzzer
- RUN/DEBUG (stato dello switch 8 di DSW1)
- Retrigger watch dog esterno
- Real time clock + 256 bytes di SRAM

Questi occupano gli indirizzi riportati nei paragrafi seguenti e non possono essere riallocati in nessun altro indirizzo. La EEPROM seriale di IC10, l'RTC e la SRAM su IC12 sono sempre gestiti dalla logica di controllo, ma effettivamente non occupano spazio d'indirizzamento in quanto sfruttano una comunicazione seriale sincrona gestita tramite linee di I/O della CPU.

MAPPAGGIO DELLE MEMORIE

Per quanto riguarda il mappaggio delle memorie, la scheda può essere configurata in 3 modi.
Di seguito viene riportata una schematizzazione di questi indirizzamenti, con le indicazioni di come devono essere settati gli switches 5 e 6 di DSW1. La scelta del mappaggio deve essere effettuata dall'utente in base al pacchetto software utilizzato e/o le richieste dell'applicazione.
Si ricorda che le posizioni degli switches non descritte é riservata per future espansioni.

MAPPAGGIO 0

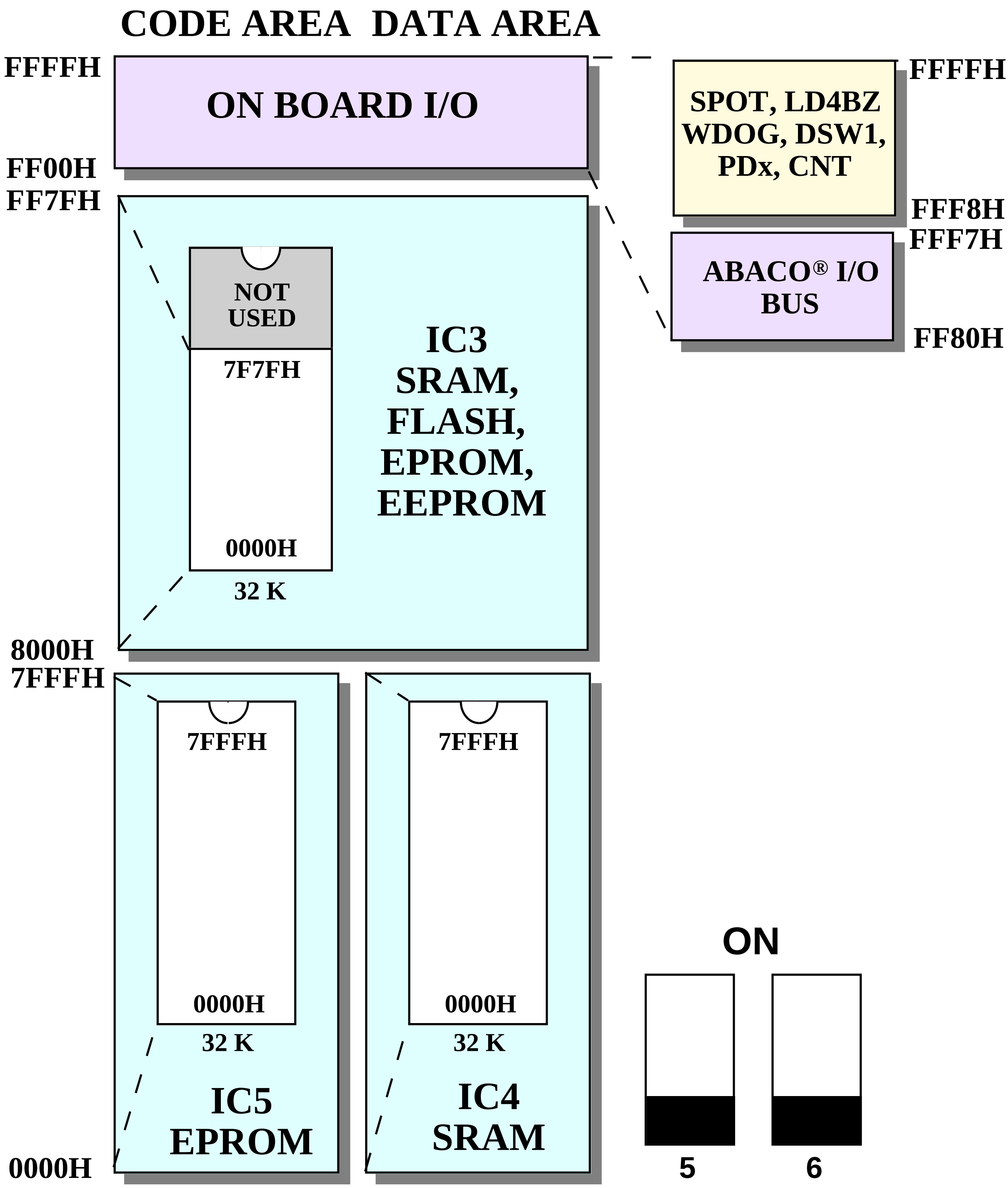


FIGURA 33: MAPPAGGIO DELLE MEMORIE IN MODO 0

Configurazione dip 5 e 6 di DSW1: **dip 5** in posizione **OFF**; **dip 6** in posizione **OFF**
Usato da pacchetti software come: BASIC 323; BXC51; HI TECH C; DDS C; ecc.

MAPPAGGIO 1

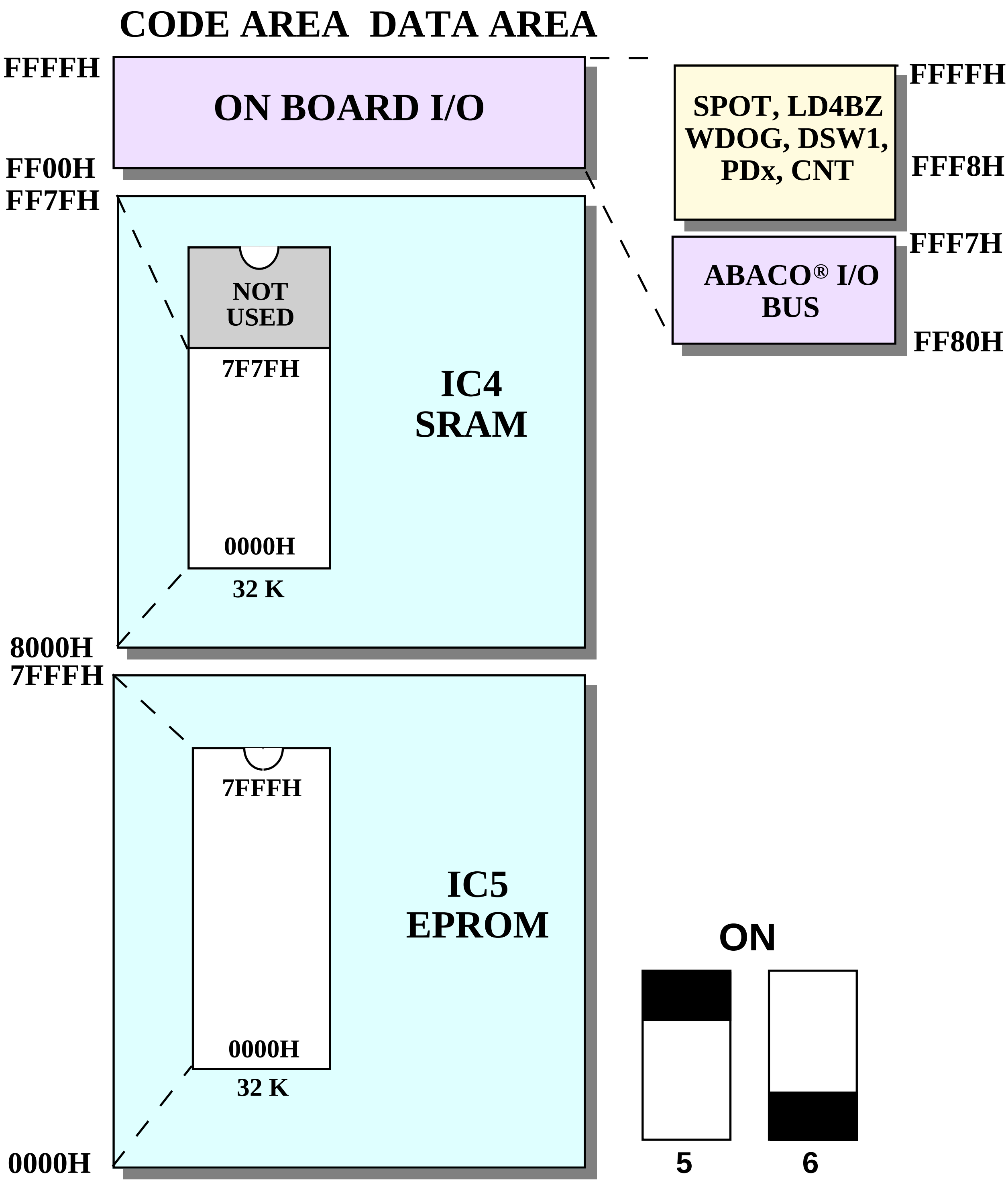


FIGURA 34: MAPPAGGIO DELLE MEMORIE IN MODO 1

Configurazione dip 5 e 6 di DSW1: **dip 5** in posizione **ON**; **dip 6** in posizione **OFF**
Usato da pacchetti software come: HI TECH C; DDS C; ecc.

MAPPAGGIO 3

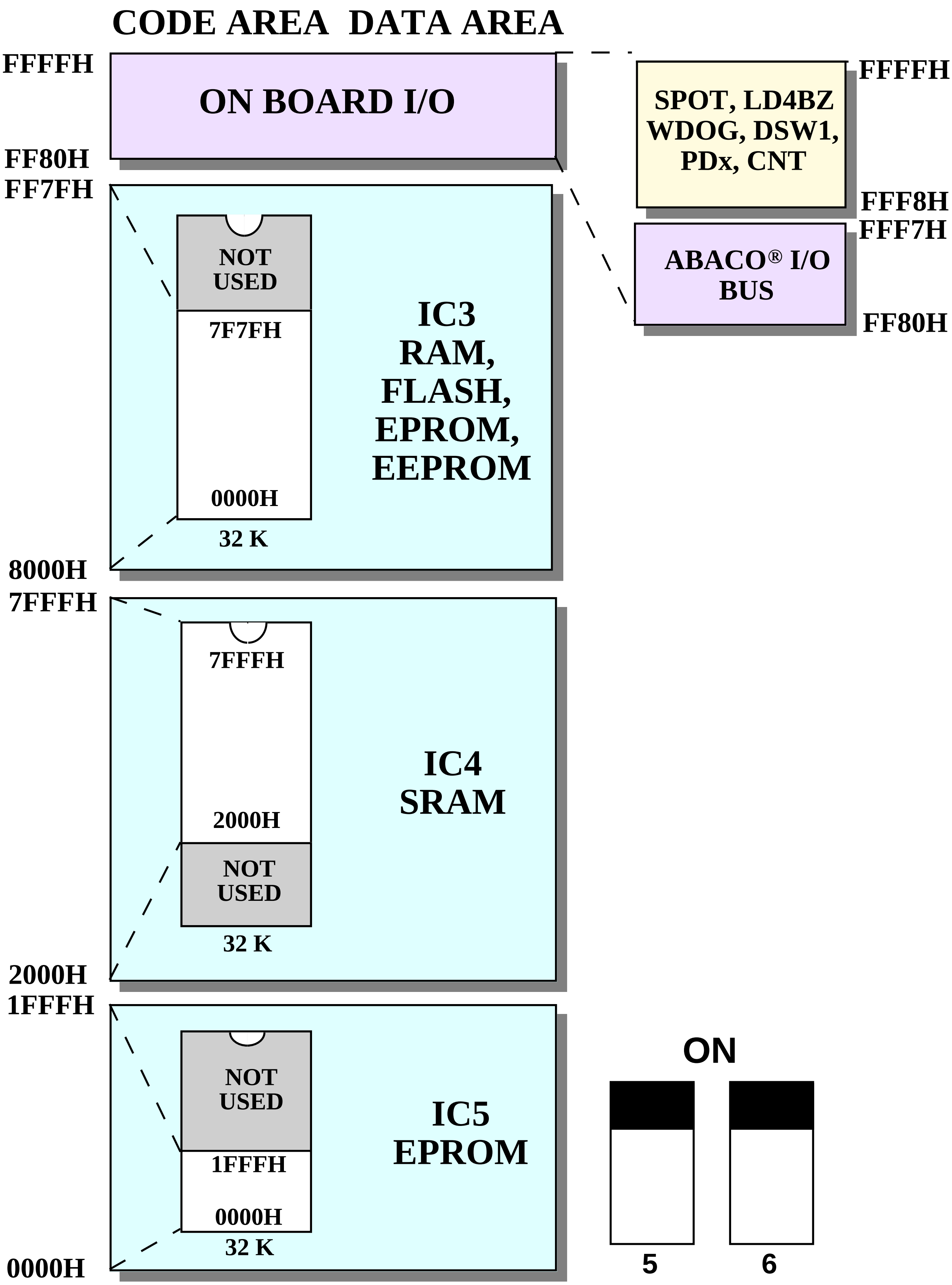


FIGURA 35: MAPPAGGIO DELLE MEMORIE IN MODO 3

Configurazione dip 5 e 6 di DSW1: **dip 5** in posizione **ON**; **dip 6** in posizione **ON**
Usato da pacchetti software come: FMO52; Lucifero; MDP; ecc.

MAPPAGGIO ABACO® I/O BUS

La logica di controllo della **GPC® 323** provvede anche alla gestione dell'**ABACO®** I/O BUS, definendo gli indirizzi in cui tale BUS viene allocato. In particolare, come si può notare dalla tabella indirizzamento I/O, tale BUS è indirizzato in corrispondenza degli indirizzi FF80H÷FFF7H. Un accesso in I/O in un qualsiasi indirizzo compreso in questi range abilita il segnale /IORQ e tutti gli altri segnali di controllo di CN1.

MAPPAGGIO DELL'I/O

Per l'I/O sono utilizzati gli ultimi 256 indirizzi (128 utilizzati per l'**ABACO® I/O BUS**, e 128 bytes per la lettura di RUN/DEBUG, il retrigger del watch dog ela gestione del real time clock) dei 64K bytes dell'area dati gestita dalla CPU. Per maggior chiarezza nella seguente tabella si riportano i nomi dei registri, i loro indirizzi, i tipi di accesso ed una breve descrizione del loro significato:

DISP.	REG.	IND.	R/W	SIGNIFICATO
ABACO® I/O BUS	I/OBUS	FF80H÷FFF7H	R/W	Indirizzi ABACO® I/O BUS
LED di spot	SPOT	FFF8H	W	Registro attivazione LED LD3 di spot
LED attività e buzzer	LD4BZ	FFF9H	W	Registro gestione LED LD4 e buzzer
Watchdog	WDOG	FFF8H	R	Registro retrigger watchdog
Dip switch	DSW1	FFF9H	R	Registro lettura DSW1 (dip 1, 2, 3, 4, 8) e stato della batteria.
PPI 82C55	PDA	FFFCH	R/W	Registro dati del port A
	PDB	FFFDH	R/W	Registro dati del port B
	PDC	FFFEH	R/W	Registro dati del port C
	CNT	FFFFH	R/W	Registro di controllo e comando

FIGURA 36: TABELLA INDIRIZZAMENTO I/O

Per quanto riguarda la descrizione del significato dei registri qui sopra riportati, si faccia riferimento al capitolo sucessivo "DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO".

DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO

Nel paragrafo precedente sono stati riportati gli indirizzi di allocazione di tutte le periferiche e di seguito viene riportata una descrizione dettagliata della funzione e del significato dei relativi registri (al fine di comprendere le successive informazioni, fare sempre riferimento alle due tabelle di indirizzamento I/O). Qualora la documentazione riportata fosse insufficiente fare riferimento direttamente alla documentazione tecnica della casa costruttrice del componente. In questo paragrafo inoltre non vengono descritte le sezioni che fanno parte del microprocessore; per quanto riguarda la programmazione di quest'ultime si faccia riferimento all'appendice B di questo manuale. Nei paragrafi successivi si usano le indicazioni D0÷D7 per fare riferimento ai bits della combinazione utilizzata nelle operazioni di I/O.

BUZZER

Il buzzer BZ1, si attiva effettuando una operazione di output con D0=1, all'indirizzo di allocazione del registro LD4BZ. Logicamente la disattivazione avviene tramite un'analoga operazione di output ma con il bit D0 resettato a 0. I rimanenti sette bit del registro LD4BZ devono essere settati tenendo conto della programmazione fornita alle circuiterie provviste di registri allocati allo stesso indirizzo, in questo caso il LED LD4. Il registro LD4BZ è azzerato (tutti i bits a 0) in fase di Reset o power on, di conseguenza in seguito ad una di queste fasi il buzzer è disattivo.

LED DI ATTIVITÀ

Il LED di attività LD4, si attiva effettuando una operazione di output con D7=1, all'indirizzo di allocazione del registro LD4BZ. Logicamente la disattivazione avviene tramite un'analoga operazione di output ma con il bit D7 resettato a 0. I rimanenti sette bit del registro LD4BZ devono essere settati tenendo conto della programmazione fornita alle circuiterie provviste di registri allocati allo stesso indirizzo, in questo caso il buzzer BZ1. Il registro LD4BZ è azzerato (tutti i bits a 0) in fase di Reset o power on, di conseguenza in seguito ad una di queste fasi il LED LD4 è disattivo.

WATCH DOG

Il retrigger della circuiteria di watch dog presente sulla **GPC® 323**, avviene tramite una semplice operazione di otput al registro WDOG. Affinché la circuiteria di watch dog non intervenga, é indispensabile retriggerarla ad intervalli regolari di durata inferiore al tempo d'intervento. Se ciò non avviene e tramite il jumper J8 la circuiteria é connessa alla sezione di reset, una volta scaduto il tempo d'intervento la scheda viene resettata. Il tempo d'intervento nella condizione di default è di circa 1,5 sec. L'intervento della circuiteria di watch dog é visualizzato dal LED LD2.

PERIFERICHE DELLA CPU

La descrizione dei registri e del relativo significato di tutte le periferiche interne della CPU (linea seriale, timer/counter, linee di I/O) é disponibile nell'appendice B.

LED DI SPOT

Il LED di spot LD3, si attiva effettuando una operazione di lettura all'indirizzo di allocazione del registro SPOT. A seguito di questa operazione il LED si attiva per un periodo di circa 300 msec e quindi si disattiva autonomamente. Il dato acquisito dall'operazione di scrittura sul registro SPOT é privo di significato e quindi può essere scartato. La funzione principale di questo LED é di segnalare il funzionamento del programma applicativo della scheda, senza appesantire lo stesso programma, oppure di segnalare visivamente il verificarsi di un evento caratterizzato da una gestione veloce.

DIP SWITCH DSW1 E RUN DEBUG

Il dip switch DSW1 montato a bordo della **GPC® 323** può essere accquisito via software, effettuando una semplice operazione di input all'indirizzo di allocazione del registro DSW1. La corrispondenza tra i bit del registro e le linee del dip switch è la seguente:

D0	->	DSW1.1
D1	->	DSW1.2
D2	->	DSW1.3
D3	->	DSW1.4
D7	->	DSW1.8

Come si nota della combinazione letta solo 5 bits riguardano l'acquisizione del dip switch, gli altri vengono usati per impostare il mappaggio di memoria (vedere paragrafo “MAPPAGGIO DELLE MEMORIE”) o sono riservati.
Il quinto dip DSW1.8 svolge la funzione di selettore delle modalità RUN o DEBUG, caratteristica di alcuni pacchetti software della **grifo®**. Si ricorda che il livello logico 0 di default corrisponde alla modalità di RUN, mentre il livello logico 1 corrisponde alla modalità di DEBUG.
L'acquisizione è in logica negata, ovvero il dip in ON fornisce lo stato logico 0 al corrispondente bit, mentre il dip in OFF fornisce lo stato logico 1.

EEPROM SERIALE

Per quanto riguarda la gestione del modulo di EEPROM seriale (IC 10), si faccia riferimento alla documentazione specifica del componente. In questo manuale tecnico non viene riportata alcuna informazione software in quanto la modalità di gestione è articolata e prevede una conoscenza approfondita del componente e comunque l'utente può usare le apposite procedure ad alto livello fornite nel pacchetto di programmazione. Si ricorda solo che i primi 32 bytes (0÷31) sono riservati e perciò si deve evitare la modifica deimedesimi. La logica di controllo della scheda consente la gestione software della EEPROM tramite due linee di I/O TTL del microcontrollore:

P3.4 (input/output)	->	linea DATA (SDA)
P1.7 (output)	->	linea CLOCK (SCL)

Data l'implementazione hardware della circuiteria di gestione del modulo di EEPROM seriale, si ricorda che i segnali A0 e A1 dello slave address sono posti a 0 logico, A2 ad 1 logico.
Lo stato logico 0 dei bit corrisponde allo stato logico basso (=0 V) del relativo segnale, mentre lo stato logico 1 dei bit corrisponde allo stato logico alto (=5 V) del relativo segnale.

SRAM TAMPONATA + RTC SERIALE

Per quanto riguarda la gestione dell'eventuale modulo di SRAM+RTC seriale (IC 12), si faccia riferimento alla documentazione specifica del componente. In questo manuale tecnico non viene riportata alcuna informazione software in quanto la modalità di gestione è articolata e prevede una conoscenza approfondita del componente e comunque l'utente può usare le apposite procedure ad alto livello fornite nel pacchetto di programmazione. Si ricorda che che in caso di montaggio di questo componente in IC 12 deve esserci al massimo una eeprom di 1024 bytes (24C08) su IC 10. Dal punto di vista elettrico la logica di controllo effettua le seguenti connessioni:

P3.4 (input/output)	->	linea DATA (SDA)
P1.7 (output)	->	linea CLOCK (SCL)

Data l'implementazione hardware della circuiteria di gestione del modulo di SRAM+RTC, si ricorda che di tale dispositivo il segnale A0 dello slave address é posto a 0.

A/D CONVERTER

Per quanto riguarda la gestione software degli 11 canali di A/D converter, di cui può essere dotata la **GPC® 323**, si faccia riferimento alla documentazione specifica del componente. In questo manuale tecnico non viene riportata alcuna informazione software in quanto la modalità di gestione è articolata e prevede una conoscenza approfondita del componente e comunque l'utente può usare le apposite procedure ad alto livello fornite nel pacchetto di programmazione. La logica di controllo della scheda consente la gestione software dell'A/D converter tramite quattro linee di I/O TTL del microcontrollore, con le seguenti corrispondenze:

P3.5 (input)	->	linea DATA OUT
P1.6 (output)	->	linea /CS
P3.4 (output)	->	linea DATA IN
P1.1 (output)	->	linea I/O CLOCK

Lo stato logico 0 dei bit corrisponde allo stato logico basso (=0 V) del relativo segnale, mentre lo stato logico 1 dei bit corrisponde allo stato logico alto (=5 V) del relativo segnale.

STATO DELLA BATTERIA

La circuiteria di back up per la SRAM su IC4 e l'RTC+SRAM su IC12 tiene continuamente sotto controllo il valore di tensione della batteria collegata e riporta su un segnale TTL l'eventualità che tale valore scenda sotto una soglia il cui valore tipico, riportato dalla documentazione del produttore del comparatore, è 2,265 Vdc. Il segnale di stato dell batteria si ottiene effettuando un accesso in lettura all'indirizzo di allocazione del registro DSW1:

D5	->	Stato della batteria

Se il bit D5 del registro DSW1 è 1 allora l'alimentazione di back-up per SRAM su IC4 e l'RTC+SRAM su IC12 è sopra 2,265 V, se è 0 allora l'alimentazione di back-up è sotto tale soglia.

PPI 82C55

Questa periferica è vista in 4 registri: uno di stato (CNT) e tre dei dati (PDA, PDB, PDC) con cui si effettua la programmazione ed il comando della stessa. I registri dati sono utilizzati sia per operazioni di lettura (acquisizione linee dei port) che per quelle di scrittura (settaggio linee dei port) ed ognuno di tali registri riporta i dati di I/O del corrispondente port. La periferica può operare in tre modi diversi:

MODO 0 = Prevede due port bidirezionali da 8 bit (A,B) e due port bidirezionali da 4 bit (C LOW, C HIGH); gli ingressi non sono latchati, mentre le uscite lo sono; nessun segnale di handshaking.

MODO 1 = Prevede due port da 12 bit (A+C LOW, B+C HIGH) dove gli 8 bit dei port A e B costituiscono le linee di I/O, mentre i 4 bit del port C costituiscono le linee di handshaking. Gli ingressi e le uscite sono latchati.

MODO 2 = Prevede un port da 13 bit (A+C3-7) dove gli 8 bit del port A costituiscono le linee di I/O, mentre i rimanenti 5 bit del port C costituiscono le linee di controllo. Un port da 11 bit (B+ C0-2) dove gli 8 bit del port B costituiscono le linee di I/O ed i rimanenti 3 bit del port C costituiscono le linee di controllo. Sia gli ingressi che le uscite sono latchate.

La programmazione della periferica avviene scrivendo un byte nel registro di controllo CNT, settando gli 8 bit del dato scritto con la seguente corrispondenza:

	D7	D6	D5	D4	D3	D2	D1	D0
CNT =	SF	M1	M2	A	CH	M3	B	CL

dove:

SF	= Se attivo (1) abilita il comando della periferica
M1 M2	= Selezionano il modo di funzionamento
0 0	= Selezione del modo 0
0 1	= Selezione del modo 1
1 X	= Selezione del modo 2
A	= Se attivo (1) setta il port A in input e viceversa
CH	= Se attivo setta il nibble più significativo del port C in input e viceversa
M3	= Se attivo (1) seleziona modo 1, viceversa seleziona modo 0
B	= Se attivo setta il port B in input e viceversa
CL	= Se attivo setta il nibble meno significativo del port C in input e viceversa.

Dopo una fase di Reset o di power on il PPI 82C55 viene settato in modo 0 con tutti i port settati in input.

DIREZIONALITÀ DELLA COMUNICAZIONE IN RS 422-485

Per gestire la direzione nella comunicazione in RS 485 o l'attivazione del driver di trasmissione nella comunicazione RS 422, sulla **GPC® 323** viene utilizzata un'apposita linea digitale di I/O del microprocessore, denominata DIR. Tale linea é collegata direttamente al pin 7 della CPU (P1.5) e come descritto nel paragrafo "SELEZIONE DEL TIPO DI COMUNICAZIONE SERIALE" ha la seguente funzione:

- RS 485:	DIR = 0	->	Linea seriale RS 485 in trasmissione
	DIR = 1	->	Linea seriale RS 485 in ricezione
- RS 422:	DIR = 0	->	Trasmettitore RS 422 attivo
	DIR = 1	->	Trasmettitore RS 422 disattivo

In fase di reset o power on, il segnale DIR é mantenuto a livello logico basso di conseguenza in seguito ad una di queste fasi il driver 485 é in ricezione o il driver di trasmissione 422 é disattivo, in modo da eliminare eventuali conflittualità sulla linea di comunicazione.

SCHEDE ESTERNE

La scheda **GPC® 323** si interfaccia a buona parte dei moduli della serie BLOCK e di interfaccia utente. Le risorse di bordo possono essere facilmente aumentate collegando la **GPC® 323** alle numerose schede periferiche del carteggio **grifo®** tramite l'**ABACO®** I/O BUS. Anche schede in formato Europa con BUS **ABACO®** possono essere collegate, sfruttando gli appositi mother boards. A titolo di esempio ne riportiamo un elenco con una breve descrizione delle carratteristiche di massima; per maggiori informazioni richiedere la documentazione specifica:

KDL X24 - KDF 224

Keyboard Display LCD 2,4 righe 24 tasti - Keyboard Display Fluorescent 2 righe 24 tasti

Interfaccia tra 16 I/O TTL su connettore normalizzato I/O **ABACO®** a 20 vie e tastiera a matrice esterna da 24 tasti; display alfanumerico fluorescente 20x 2 o LCD 20x2, 20x4 retroilluminato a LEDs. Predisposizione per collegamento a tastiera telefonica.

QTP 24P

Quick Terminal Panel 24 tasti con interfaccia Parallela

Interfaccia operatore provvista di display alfanumerico fluorescente 20x 2 o LCD 20x2, 20x4 retroilluminato a LEDs; tastiera a membrana da 24 tasti di cui 12 configurabili dall'utente; 16 LEDs di stato; alimentatore a bordo scheda in grado di pilotare anche carichi esterni; interdaccia parallela basata su 16 I/O TTL di un connettore normalizzato I/O **ABACO®** a 20 vie. Tasti ed etichette personalizzabili tramite serigrafie da inserire in apposite tasche; opzione di contenitore metallico.

QTP G28

Quick Terminal Panel 28 tasti con LCD grafico

Interfaccia operatore provvista di display grafico da 240x128 pixel retroilluminato a catodo freddo; tastiera a membrana da 28 tasti di cui 6 configurabili dall'utente; 16 LEDs di stato; alimentatore a bordo scheda; interdaccia seriale in RS 232, RS 422-485 o current loop; linea seriale ausiliaria in RS 232 Tasti e LED personalizzabili tramite serigrafie da inserire in apposite tasche; ; EEPROM di set up; 256K EPROM o FLASH; Real Time Clock; 128K RAM; buzzer; interfaccia CAN; relé utente. Firmware di gestione che svolge funzione di terminale con primitive grafiche.

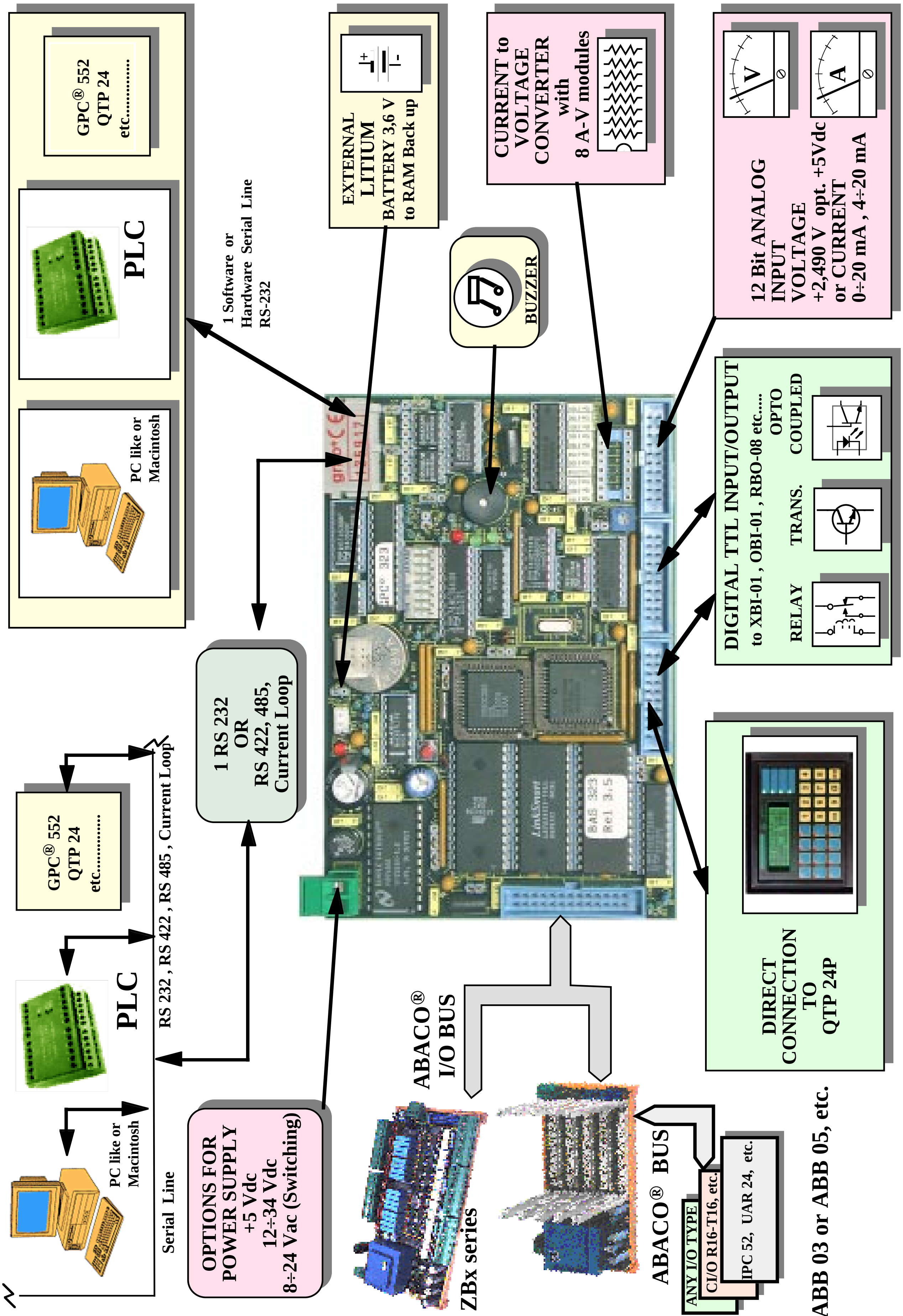


FIGURA 36: SCHEMA DELLE POSSIBILI CONNESSIONI DELLA GPC®323

MCI 64
Memory Cards Interfaces 64 MBytes
Interfaccia per la gestione di Memory cards PCMCIA a 68 pins tramite un connettore normalizzato I/O **ABACO®**; sono disponibili driver per linguaggi ad alto livello.

IBC 01
Interface Block Comunication
Scheda di conversioni per comunicazioni seriali. 2 linee RS 232; 1 linea RS 422-485; 1 linea in fibra ottica; interfaccia DTE/DCE selezionabile; attacco rapido per guide tipo DIN 46277-1 e 3.

ZBR xxx
Zipped BLOCK Relays xx Input + xx Output
Periferica per xx Input optoisolati e visualizzati tipo NPN; xx relé da 3A con MOV; connettori a morsettiera per ingressi optoisolati e uscite; connettore normalizzato **ABACO®** I/O BUS; LEDs di visualizzazione; sezione alimentatrice a bordo; attacco rapide per guide **Ω**. Le possibili configurazioni in termini di numero di I/O sono: xxx=324 con 32 In e 24 Out; xxx=246 con 24 In e 16 Out; xxx=168 con 16 In e 8 Out; xxx=84 con 8 In e 4 Out.

IAC 01
Interface Adapter Centronics
Interfaccia tra 16 I/O TTL su connettore normalizzato I/O **ABACO®** a 20 vie e connettore a vaschetta D 25 vie femmina con pin out standard Centronics per la gestione di una stampante parallela.

OBI N8 - OBI P8
Opto BLOCK Input NPN-PNP
Interfaccia per 8 input optoisolati e visualizzati tipo NPN, PNP, connettore a morsettiera, connettore normalizzato I/O **ABACO®** a 20 vie; sezione alimentatrice; attacco rapido per guide DIN 46277-1 e 3.

TBO 01 - TBO 08
Transistor BLOCK Output
Interfaccia per 16 connettore normalizzato I/O **ABACO®** a 20 vie; 16 o 8 output a transistor in Open Collector da 45 Vcc 3 A su connettore a morsettiera. Uscite optoisolate e visualizzate; attacco rapido per guide DIN 6277-1 e 3.

RBO 08 - RBO 16
Relé BLOCK Output
Interfaccia per connettore normalizzato I/O **ABACO®** a 20 vie; 8 o 16 output visualizzati con relé da 3 A con MOV; connettore a morsettiera; attacco rapido per guide DIN 46277-1 e 3.

FBC 20 - FBC 120
Flat Block Contact 20 vie
Interfaccia tra 2 o 1 connettori a perforazione di isolante (scatolino da 20 vie maschi) e la filatura da campo (morsettiere a rapida estrazione). Attacco rapido per guide tipo DIN 46277-1 e 3.

DEB 01
Didactis Experimental Board
Scheda di supportro per l'utilizzo di 16 linee di I/O TTL. Comprende: 16 tasti; 16 LED; 4 digits; tastiera a matrice da 16 tasti; interfaccia per stampante Centronics, dislay LCD, display Fluorescente, connettore I/O **GPC® 68**; collegamento con il campo.

IAL 42

Interface Adapter LCD

Interfaccia tra 16 I/O TTL su connettore normalizzato I/O **ABACO®** e connettore a scatolino a 14 vie con pin out standard per la gestione di display fluorescenti LCD.

XBI 01

miXed BLOCK Input-Output

Interfaccia tra 8 input + 8 output TTL (connettore normalizzato I/O **ABACO®** a 20 vie), con 8 output a transistor in Open Collector da 45 Vcc 3 A + 8 input con filtro a Pi-Greco (connettore a morsettiera). I/O optoisolati e visualizzati; attacco rapido per guide DIN 46277-1 e 3.

XBI R4 - XBI T4

miXed BLOCK Input-Output

Interfaccia per connettore normalizzato I/O **ABACO®** a 20 vie; 4 relé da 3 A con MOV o 4 transistor open collectors da 3 A optoisolati; 4 linee di input optoisolate; linee di I/O visualizzate; connettore a morsettiera; attacco rapido per guide DIN tipo C e guide Ω.

ZBT xxx

Zipped BLOCK Transistors xx Input + xx Output

Periferica per xy Input optoisolati e visualizzati tipo NPN; yz darlinghton da 3A con diodo di ricircolo; connettori a morsettiera per ingressi optoisolati e uscite; connettore normalizzato **ABACO®** I/O BUS; 61 LEDs di visualizzazione; sezione alimentatrice a bordo; attacco rapido per guide Ω. Le possibili configurazioni in termini di numero di I/O sono: xxx=324 con 32 In e 24 Out; xxx=246 con 24 In e 16 Out; xxx=168 con 16 In e 8 Out; xxx=84 con 8 In e 4 Out.

ABB 05

Abaco® Block BUS 5 slots

Mother board **ABACO®** da 5 slots; passo 4 TE; guidaschede; connettori normalizzati di alimentazione; tasto di reset; LEDs per alimentazioni; interfaccia **ABACO®** I/O BUS; sezione alimentatrice per +5 Vdc; sezione alimentatrice per +V Opto; sezioni alimentatrici galvanicamente isolate; tre tipi di alimentazione: da rete, bassa tensione o stabilizzata. Attacco rapido per guide Ω.

ABB 03

Abaco® Block BUS 3 slots

Mother board ABACO® da 3 slots; passo 4 TE; guidaschede; connettori normalizzati di alimentazione; tasto di reset; LEDs per alimentazioni; interfaccia **ABACO®** I/O BUS. Attacco rapido per guide Ω.

BIBLIOGRAFIA

E' riportato di seguito, un elenco di manuali e note tecniche, a cui l'utente può fare riferimento per avere maggiori chiarimenti, sui vari componenti montati a bordo della scheda **GPC® 323**.

Manuale TEXAS INSTRUMENTS:	*The TTL Data Book - SN54/74 Families*
Manuale TEXAS INSTRUMENTS:	*RS-422 and RS-485 Interface Circuits*
Manuale TEXAS INSTRUMENTS:	*Linear Circuits Dtata Book - Volumi 1 e 3*
Manuale TEXAS INSTRUMENTS:	*Data Acquisition Circuits Data Book*
Manuale NEC:	*Microprocessors and Peripherals - Volume 3*
Manuale NEC:	*Memory Products*
Manuale HEWLETT PACKARD:	*Optoelectronics Designer's Catalog*
Manuale MAXIM:	*New Releases Data Book - Volume 4*
Manuale MAXIM:	*New Releases Data Book - Volume 5*
Manuale XICOR:	*Data Book*
Manuale PHILIPS:	*80C51 - Based 8-Bit Microcontrollers*
Manuale PHILIPS:	*80C51 - 8 bit Flash microcontroller family*
Manuale PHILIPS:	*I^2C bus*
Manuale DALLAS SEMICONDUCTOR:	*1992-1993 Product Data Book SUPPLEMENT*
Manuale NATIONAL SEMICONDUCTOR:	*Linear Databook - Volume 1*
Manuale INTEL:	*8 Bit Embedded Microcontrollers*
Manuale TOSHIBA:	*Mos Memory Products*

Per ulteriori informazioni e aggiornamenti si prega di fare riferimento alle pagine Internet specifiche gestite dalle compagnie costruttrici.

APPENDICE A: SCHEMI ELETTRICI

In questa appendice sono disponibili gli schemi elettrici delle interfacce per la **GPC® 323** più frequentemente utilizzate. Tutte queste interfaccie possono essere prodotte autonomamente dall'utente mentre solo alcune di esse sono schede **grifo®** standard e possono quindi essere ordinate.

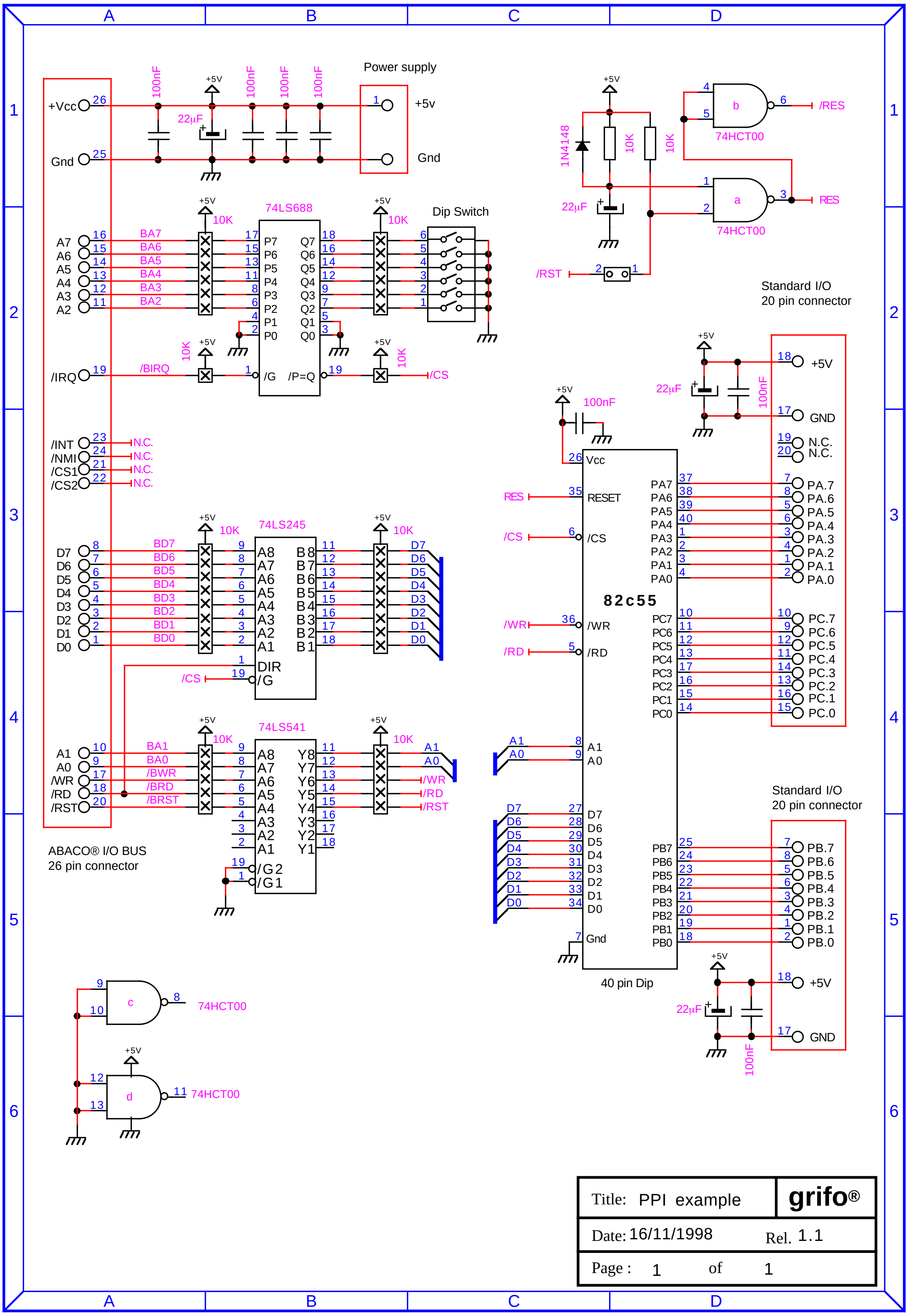


FIGURA A1: SCHEMA ELETTRICO DI ESPANSIONE PPI

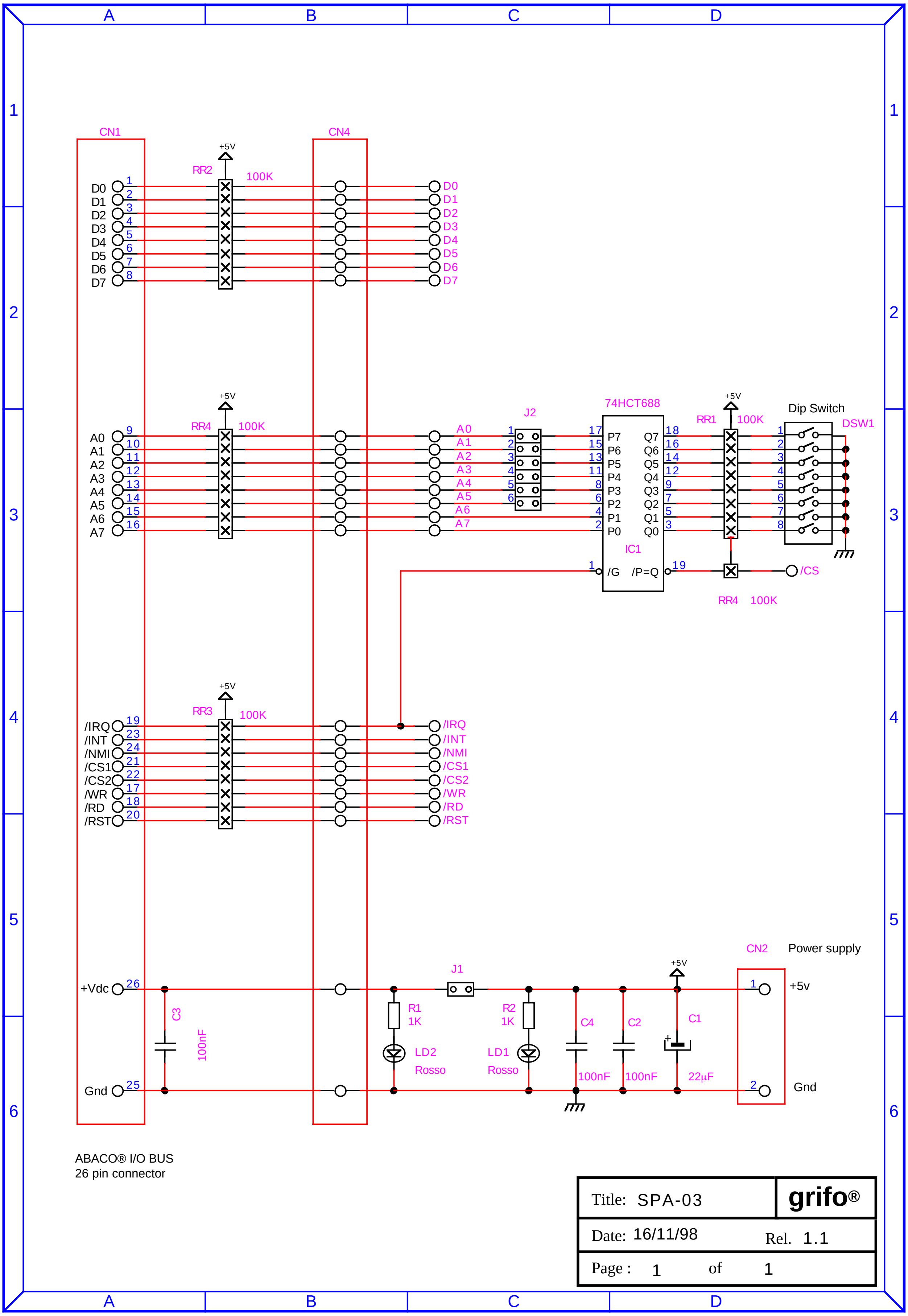


FIGURA A2: SCHEMA ELETTRICO SPA 03

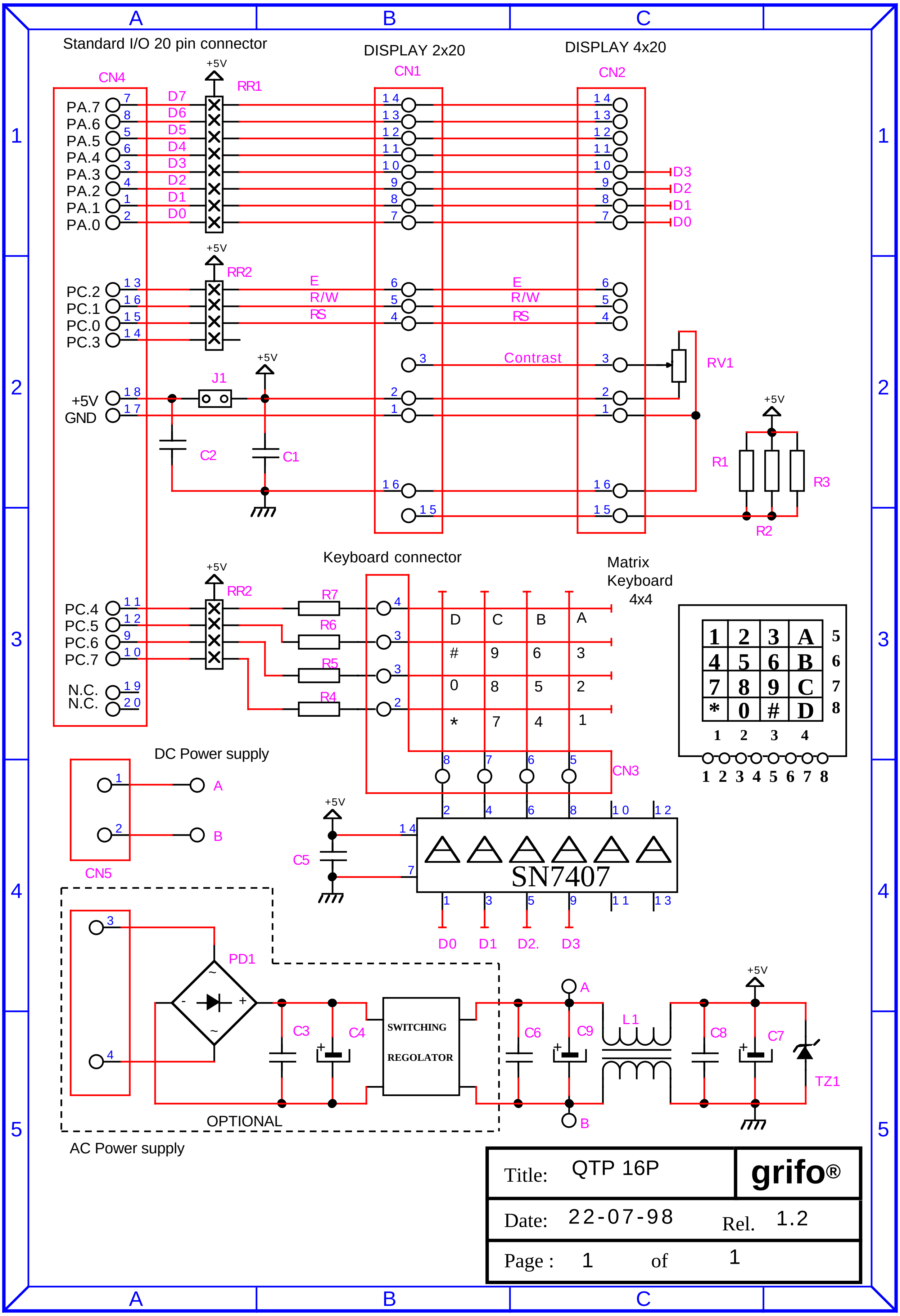


FIGURA A3: SCHEMA ELETTRICO QTP 16P

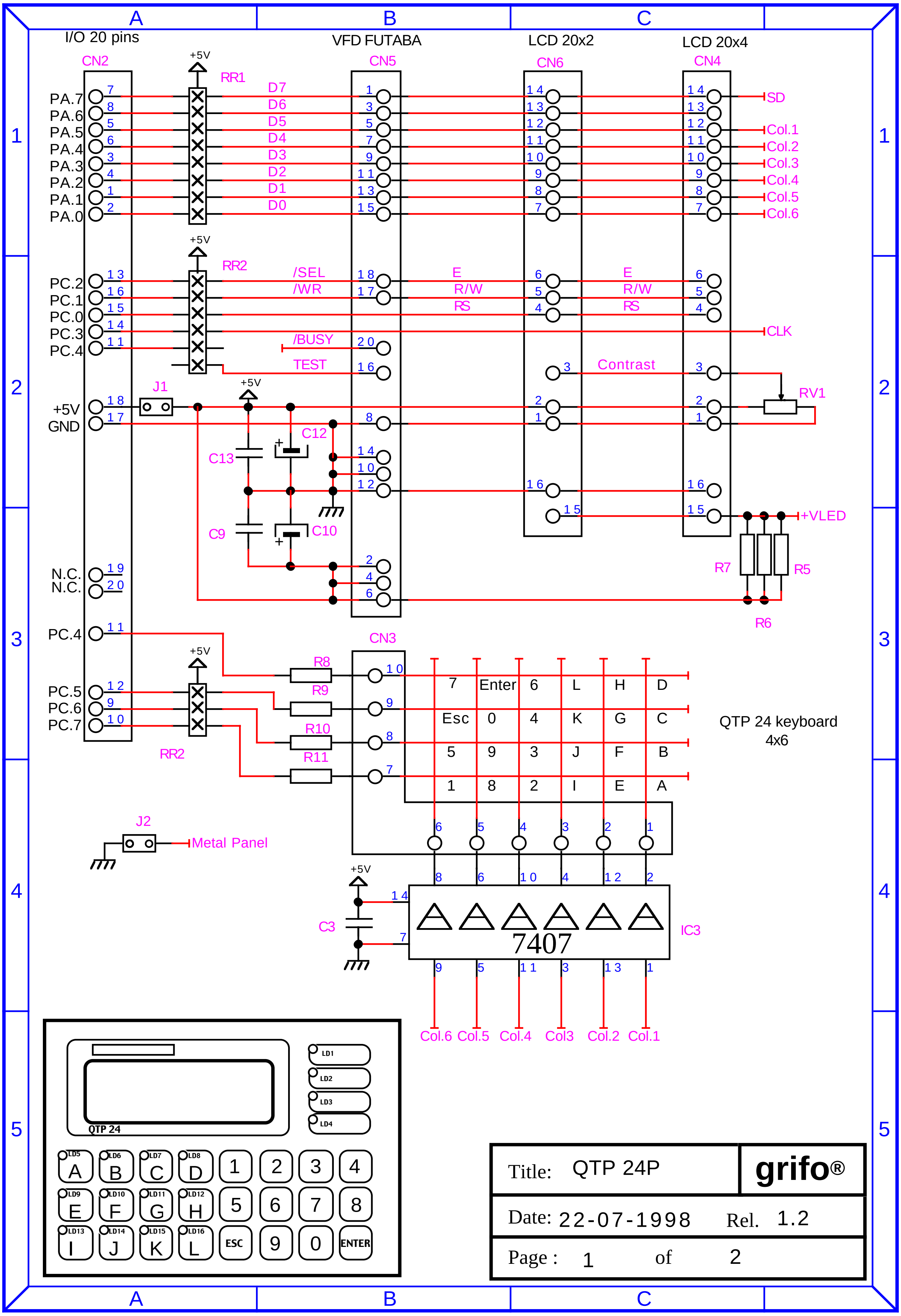


FIGURA A4: SCHEMA ELETTRICO QTP 24P 1/2

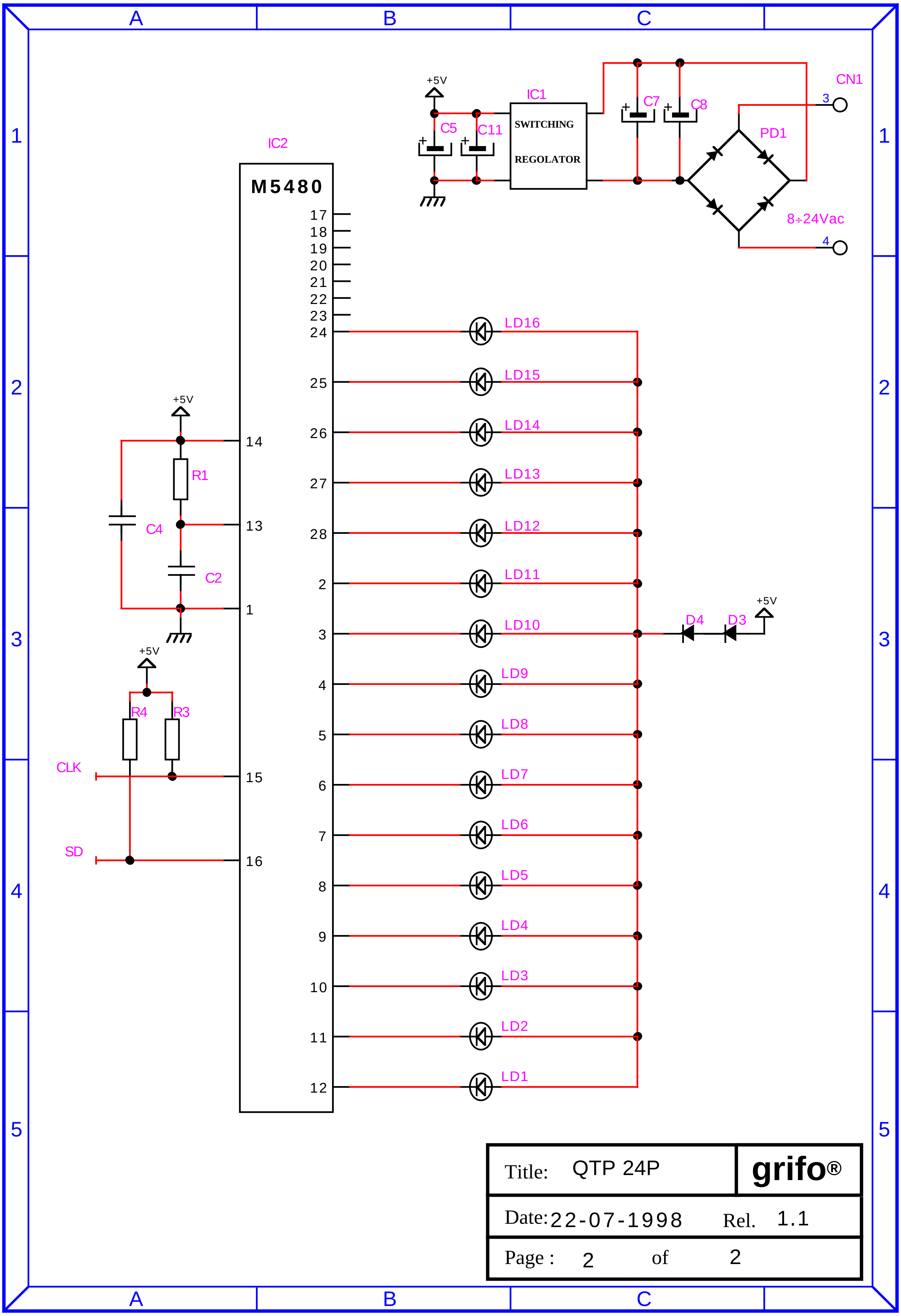


FIGURA A5: SCHEMA ELETTRICO QTP 24P 2/2

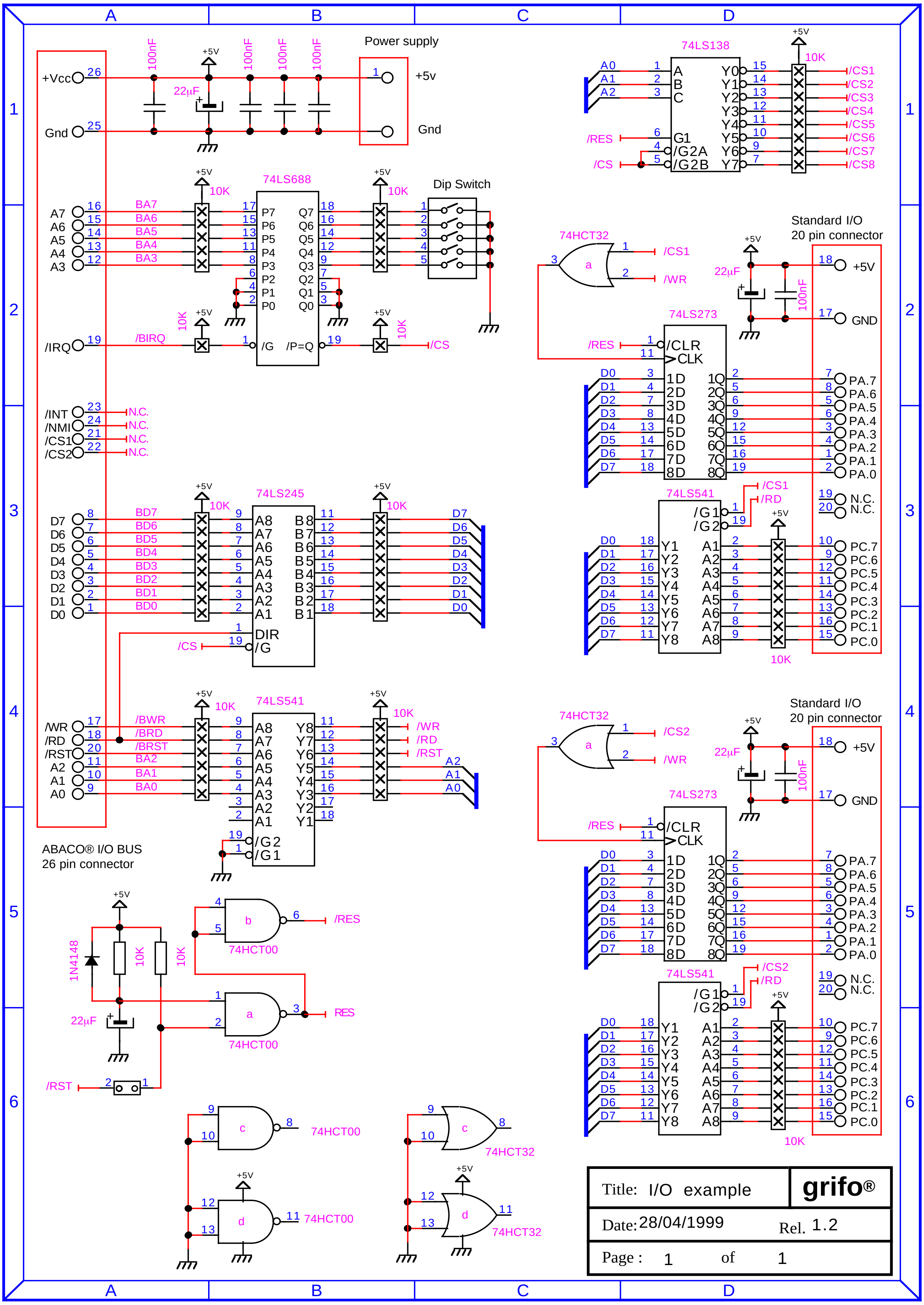


FIGURA A6: SCHEMA ELETTRICO DI I/O SU ABACO® I/O BUS

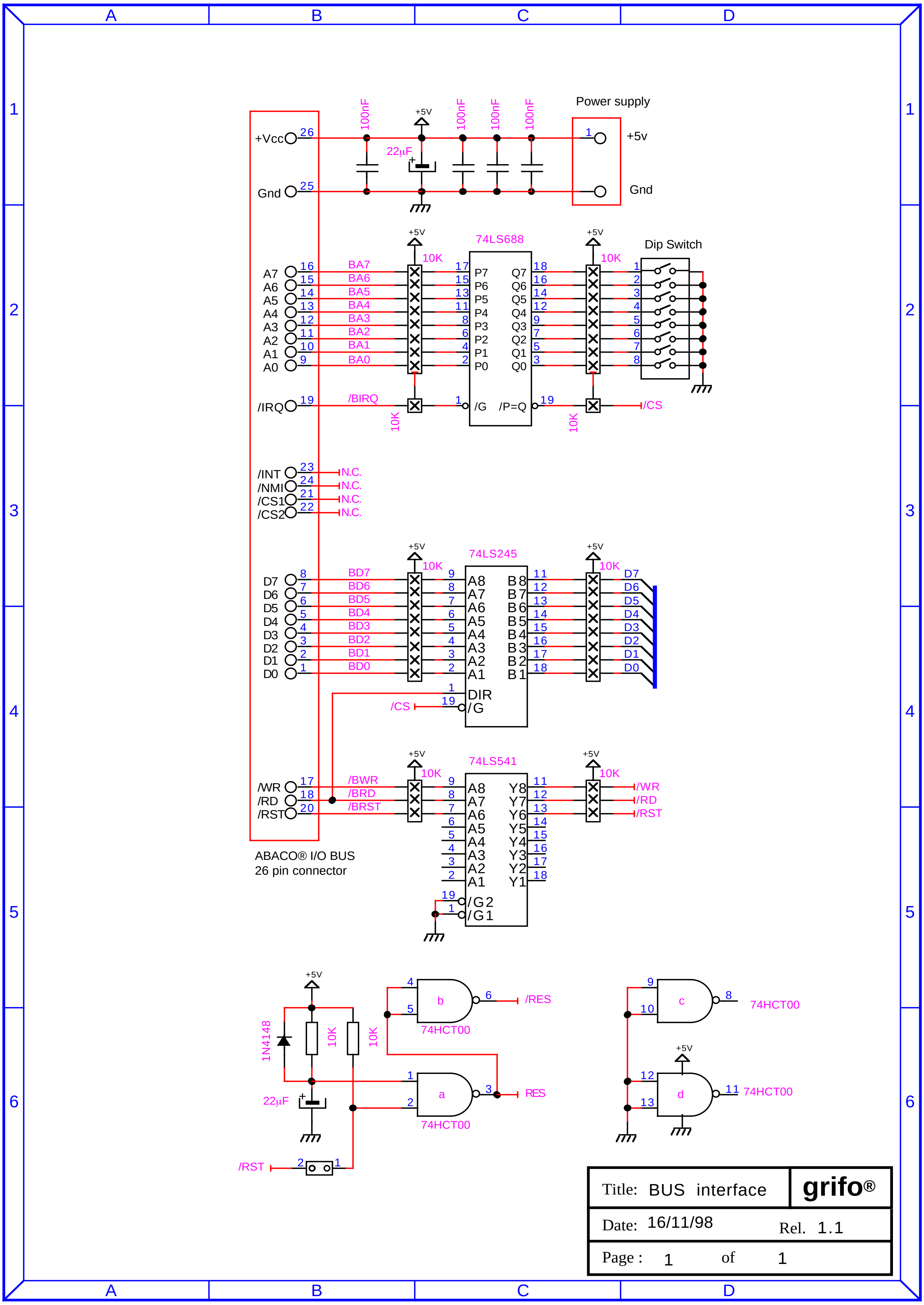


FIGURA A7: SCHEMA ELETTRICO INTERFACCIA BUS

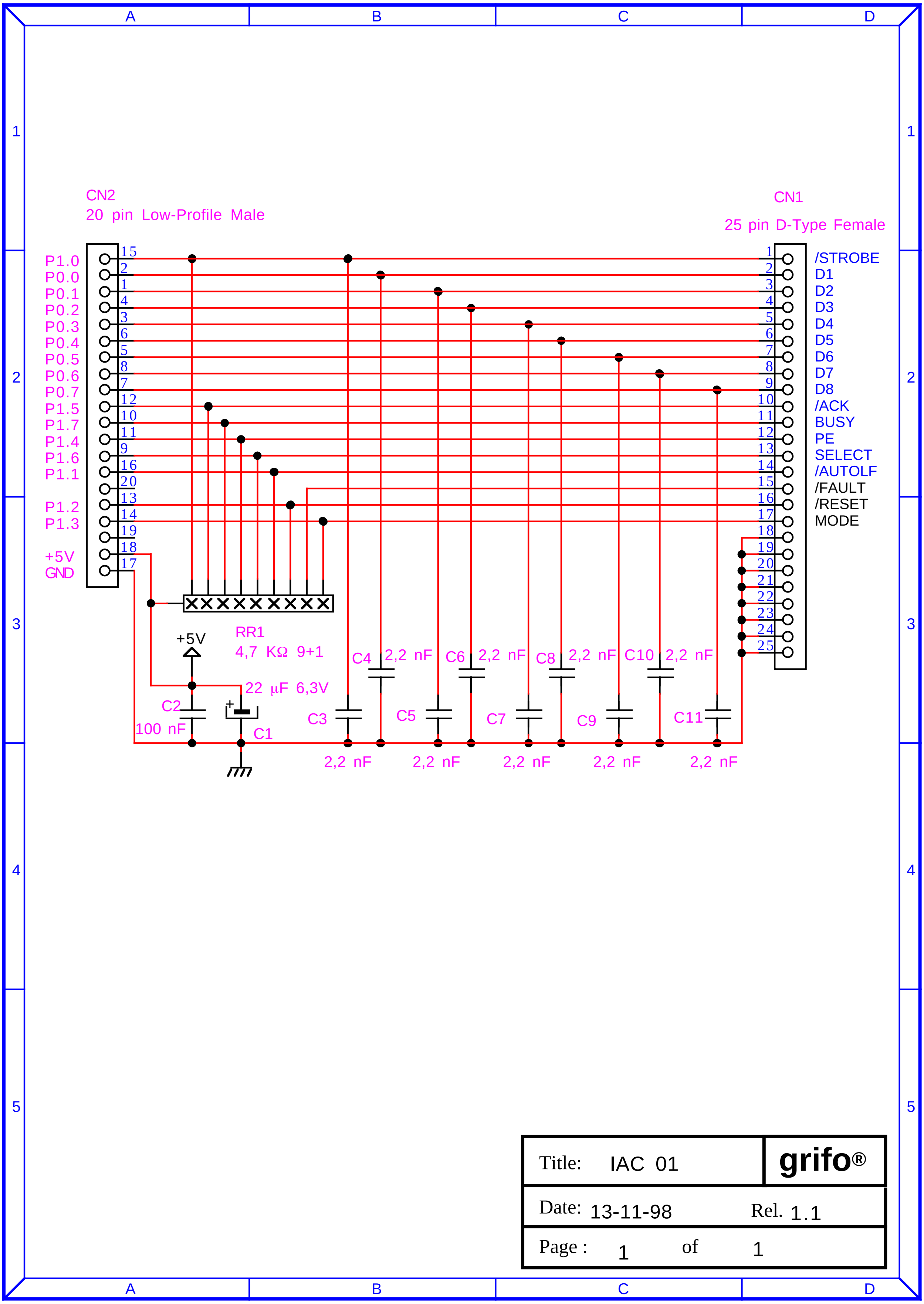


FIGURA A8: SCHEMA ELETTRICO IAC 01

CMOS single-chip 8-bit microcontrollers 80C32/87C52

DESCRIPTION

The Philips 80C32/87C52 is a high-performance microcontroller fabricated with Philips high-density CMOS technology. The Philips CMOS technology combines the high speed and density characteristics of HMOS with the low power attributes of CMOS. Philips epitaxial substrate minimizes latch-up sensitivity.

The 87C52 contains an 8k × 8 EPROM and the 80C32 is ROMless. Both contain a 256 × 8 RAM, 32 I/O lines, three 16-bit counter/timers, a six-source, two-priority level nested interrupt structure, a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART, and on-chip oscillator and clock circuits.

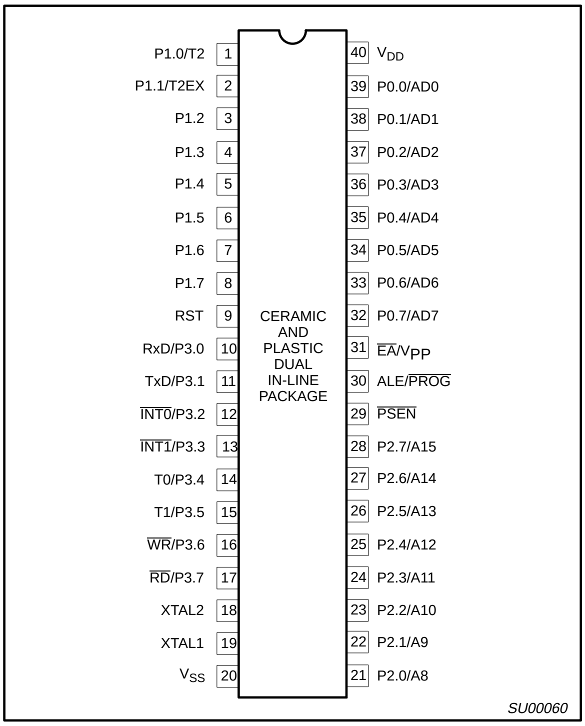
In addition, the 80C32/87C52 has two software selectable modes of power reduction—idle mode and power-down mode. The idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

See 80C52/80C54/80C58 datasheet for ROM device specifications.

FEATURES

- 80C51 based architecture
- 8032 compatible
  - 8k × 8 EPROM (87C52)
  - ROMless (80C32)
  - 256 × 8 RAM
  - Three 16-bit counter/timers
  - Full duplex serial channel
  - Boolean processor
- Memory addressing capability
  - 64k ROM and 64k RAM
- Power control modes:
  - Idle mode
  - Power-down mode
- CMOS and TTL compatible
- Three speed ranges:
  - 3.5 to 16MHz
  - 3.5 to 24MHz
  - 3.5 to 33MHz
- Five package styles
- Extended temperature ranges
- OTP package available

PIN CONFIGURATIONS



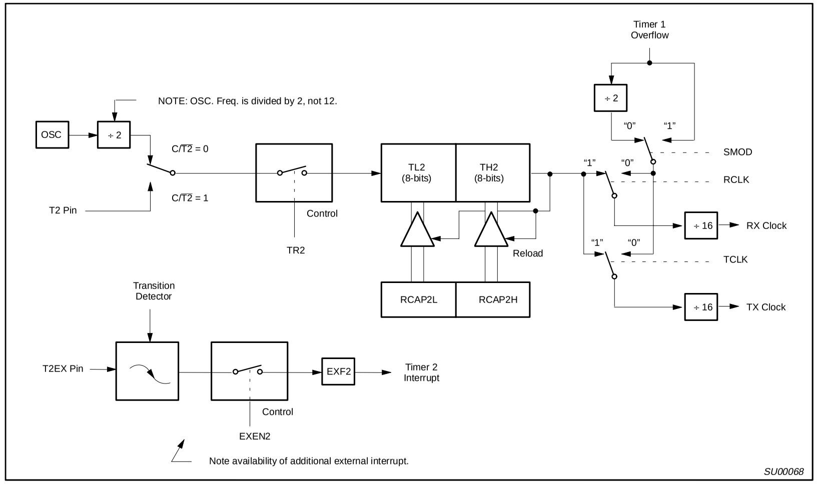


Figure 4. Timer 2 in Baud Rate Generator Mode

Table 2. Timer 2 Operating Modes

RCLK + TCLK	CP/RL2	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	X	1	Baud rate generator
X	X	0	(off)

Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK + TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Therefore, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt, if desired.

It should be noted that when Timer 2 is running (TR2 = 1) in "timer" function in the baud rate generator mode, one should not try to read or write TH2 or TL2. Under these conditions the timer is being incremented every state time, and the results of a read or write may not be accurate. The RCAP registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. Turn the timer off (clear TR2) before accessing the Timer 2 or RCAP registers, in this case.

Timer/Counter 2 Set-up

Except for the baud rate generator mode, the values given for T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set, separately, to turn the timer on. See Table 3 for set-up of timer 2 as a timer. See Table 4 for set-up of timer 2 as a counter.

Using Timer/Counter 2 to Generate Baud Rates

For this purpose, Timer 2 must be used in the baud rate generating mode. If Timer 2 is being clocked through pin T2 (P1.0) the baud rate is:

$$\text{Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

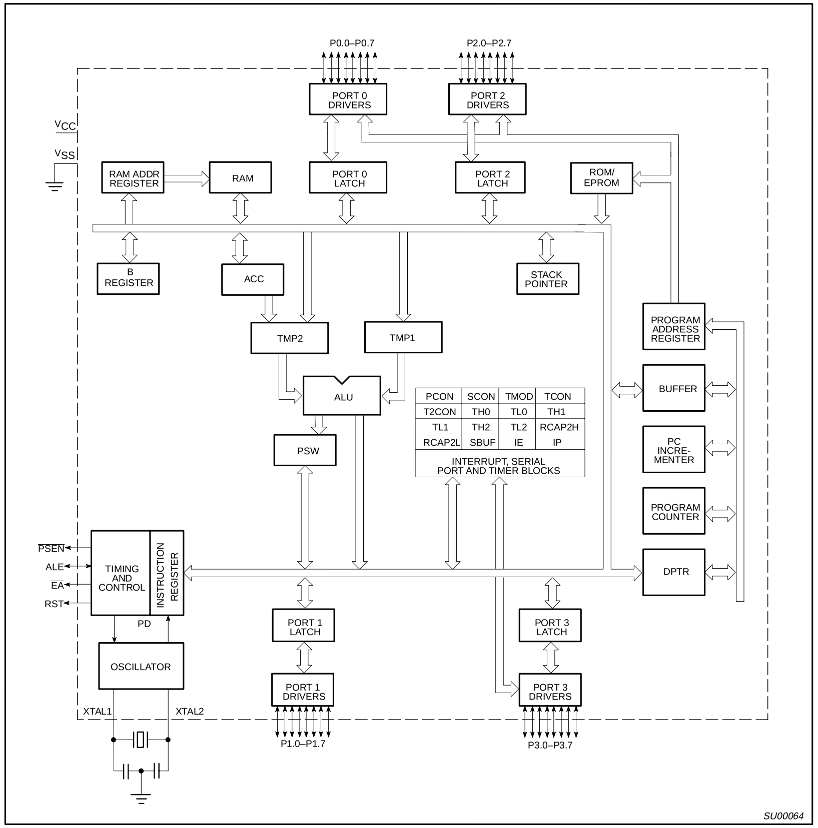
And if it is being clocked internally, the baud rate is:

$$\text{Baud Rate} = \frac{\text{Oscillator Frequency}}{32 \times [65536 - (\text{RCAP2H, RCAP2L})]}$$

To obtain the reload value for RCAP2H and RCA02L, the above equation can be rewritten as:

$$\text{RCAP2H, RCAP2L} = 65536 - \left(\frac{\text{Oscillator Frequency}}{32 \times \text{Baud Rate}}\right)$$

BLOCK DIAGRAM



CMOS single-chip 8-bit microcontrollers 80C32/87C52

Table 1. 8XC52 Special Function Registers

SYMBOL	DESCRIPTION	DIRECT ADDRESS	MSB							LSB	RESET VALUE
			BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION								
ACC*	Accumulator	E0H	E7	E6	E5	E4	E3	E2	E1	E0	00H
B*	B register	F0H	F7	F6	F5	F4	F3	F2	F1	F0	00H
DPTR:	Data pointer (2 bytes)										
DPH	Data pointer high	83H									00H
DPL	Data pointer low	82H									00H
			AF	AE	AD	AC	AB	AA	A9	A8	
IE*	Interrupt enable	A8H	EA	–	ET2	ES	ET1	EX1	ET0	EX0	0x000000B
			BF	BE	BD	BC	BB	BA	B9	B8	
IP*	Interrupt priority	B8H	–	–	PT2	PS	PT1	PX1	PT0	PX0	xx000000B
			87	86	85	84	83	82	81	80	
P0*	Port 0	80H	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	FFH
			97	96	95	94	93	92	91	90	
P1*	Port 1	90H	–	–	–	–	–	–	T2EX	T2	FFH
			A7	A6	A5	A4	A3	A2	A1	A0	
P2*	Port 2	A0H	A15	A14	A13	A12	A11	A10	A9	A8	FFH
			B7	B6	B5	B4	B3	B2	B1	B0	
P3*	Port 3	B0H	$\overline{RD}$	$\overline{WR}$	T1	T0	$\overline{INT1}$	$\overline{INT0}$	TxD	RxD	FFH
PCON[1]	Power control	87H	SMOD	–	–	–	GF1	GF0	PD	IDL	0xxxxxxxB
			D7	D6	D5	D4	D3	D2	D1	D0	
PSW*	Program status word	D0H	CY	AC	F0	RS1	RS0	OV	–	P	00H
RCAP2H#	Capture high	CBH									00H
RCAPL#	Capture low	CAH									00H
SBUF	Serial data buffer	99H									xxxxxxxxB
			9F	9E	9D	9C	9B	9A	99	98	
SCON*	Serial controller	98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00H
SP	Stack pointer	81H									07H
			8F	8E	8D	8C	8B	8A	89	88	
TCON*	Timer control	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00H
			CF	CE	CD	CC	CB	CA	C9	C8	
T2CON*#	Timer 2 control	C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$	00H
TH0	Timer high 0	8CH									00H
TH1	Timer high 1	8DH									00H
TH2#	Timer high 2	CDH									00H
TL0	Timer low 0	8AH									00H
TL1	Timer low 1	8BH									00H
TL2#	Timer low 2	CCH									00H
TMOD	Timer mode	89H	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0	00H

* Bit addressable
\# SFRs are modified from or added to the 80C51 SFRs.
1. Bits GF1, GF0, PD, and IDL of the PCON register are not implemented in the NMOS 8XC52.

CMOS single-chip 8-bit microcontrollers 80C32/87C52

PIN DESCRIPTION

MNEMONIC	PIN NO. DIP	PIN NO. LCC	PIN NO. QFP	TYPE	NAME AND FUNCTION
V_{SS}	20	22	16	I	**Ground:** 0V reference.
V_{CC}	40	44	38	I	**Power Supply:** This is the power supply voltage for normal, idle, and power-down operation.
P0.0–0.7	39–32	43–36	37–30	I/O	**Port 0:** Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pull-ups when emitting 1s. Port 0 also outputs the code bytes during program verification in the 87C52. External pull-ups are required during program verification.
P1.0–P1.7	1–8	2–9	40–44 1–3	I/O	**Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: I_{IL}). Pins P1.0 and P1.1 also. Port 1 also receives the low-order address byte during program memory verification. Port 1 also serves alternate functions for timer 2:
	1	2	40	I	**T2 (P1.0):** Timer/counter 2 external count input.
	2	3	41	I	**T2EX (P1.1):** Timer/counter 2 trigger input.
P2.0–P2.7	21–28	24–31	18–25	I/O	**Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: I_{IL}). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOV @Ri), port 2 emits the contents of the P2 special function register.
P3.0–P3.7	10–17	11, 13–19	5, 7–13	I/O	**Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: I_{IL}). Port 3 also serves the special features of the 80C51 family, as listed below:
	10	11	5	I	**RxD (P3.0):** Serial input port
	11	13	7	O	**TxD (P3.1):** Serial output port
	12	14	8	I	**$\overline{INT0}$ (P3.2):** External interrupt
	13	15	9	I	**$\overline{INT1}$ (P3.3):** External interrupt
	14	16	10	I	**T0 (P3.4):** Timer 0 external input
	15	17	11	I	**T1 (P3.5):** Timer 1 external input
	16	18	12	O	**$\overline{WR}$ (P3.6):** External data memory write strobe
	17	19	13	O	**$\overline{RD}$ (P3.7):** External data memory read strobe
RST	9	10	4	I	**Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to V_{SS} permits a power-on reset using only an external capacitor to V_{CC}.
ALE/$\overline{PROG}$	30	33	27	I/O	**Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{PROG}$) during EPROM programming.
$\overline{PSEN}$	29	32	26	O	**Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory.
$\overline{EA}$/V_{PP}	31	35	29	I	**External Access Enable/Programming Supply Voltage:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to 1FFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 1FFFH. This pin also receives the 12.75V programming supply voltage (V_{PP}) during EPROM programming.
XTAL1	19	21	15	I	**Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits.
XTAL2	18	20	14	O	**Crystal 2:** Output from the inverting oscillator amplifier.

DIFFERENCES FROM THE 80C51

Special Function Registers

The special function register space is the same as the 80C51 except that the 80C32/87C52 contains the additional special function registers T2CON, RCAP2L, RCAP2H, TL2, and TH2. Since the standard 80C51 on-chip functions are identical in the 8XC52, the SFR locations, bit locations, and operation are likewise identical. The only exceptions are in the interrupt mode and interrupt priority SFRs (see Table 1).

Timer/Counters

In addition to timer/counters 0 and 1 of the 80C51, the 80C32/87C52 contains timer/counter 2. Like timers 0 and 1, timer 2 can operate as either an event timer or as an event counter. This is selected by bit C/T2 in the special function register T2CON (see Figure 1). It has three operating modes: capture, auto-load, and baud rate generator, which are selected by bits in the T2CON as shown in Table 2.

In the Capture Mode there are two options which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter which upon overflowing sets bit TF2, the Timer 2 overflow bit, which can be used to generate an interrupt. If EXEN2 = 1, then Timer 2 still does the above, but with the added feature that a 1-to-0 transition at external input T2EX causes the current value in the Timer 2 registers, TL2 and TH2, to be captured into registers RCAP2L and RCAP2H, respectively. (RCAP2L and RCAP2H are new special function registers in the 80C52.) In addition, the transition at T2EX causes bit EXF2 in T2CON to be set, and EXF2 like TF2 can generate an interrupt. The Capture Mode is illustrated in Figure 2.

In the auto-reload mode, there are again two options, which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then when Timer 2 rolls over it not only sets TF2 but also causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2L and RCAP2H, which are preset by software. If EXEN2 = 1, then Timer 2 still does the above, but with the added feature that a 1-to-0 transition at external input T2EX will also trigger the 16-bit reload and set EXF2. The auto-reload mode is illustrated in Figure 3.

The baud rate generation mode is selected by RCLK = 1 and/or TCLK = 1. It will be described in conjunction with the serial port.

Serial Port

The serial port of the 8XC52 is identical to that of the 80C51 except that counter/timer 2 can be used to generate baud rates.

In the 8XC52, Timer 2 is selected as the baud rate generator by setting TCLK and/or RCLK in T2CON (see Figure 1). Note that the baud rate for transmit and receive can be simultaneously different. Setting RCLK and/or TCLK puts Timer into its baud rate generator mode, as shown in Figure 4.

The baud rate generator mode is similar to the auto-reload mode, in that a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

Now, the baud rates in Modes 1 and 3 are determined by Timer 2's overflow rate as follows:

$$\text{Modes 1, 3 Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

The timer can be configured for either "timer" or "counter" operation. In the most typical applications, it is configured for "timer" operation (C/T2 = 0). "Timer" operation is a little different for Timer 2 when it's being used as a baud rate generator. Normally, as a timer it would increment every machine cycle (thus at 1/12 the oscillator frequency). As a baud rate generator, however, it increments every state time (thus at 1/2 the oscillator frequency). In that case the baud rate is given by the formula:

$$\text{Modes 1, 3 Baud Rate} = \frac{\text{Oscillator Frequency}}{32 \quad [65536 - (RCAP2H, RCAP2L)]}$$

where (RCAP2H, RCAP2L) is the content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

(MSB)							(LSB)
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$

Symbol	Position	Name and Significance
TF2	T2CON.7	Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK or TCLK = 1.
EXF2	T2CON.6	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software.
RCLK	T2CON.5	Receive clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.
TCLK	T2CON.4	Transmit clock flag. When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EXEN2	T2CON.3	Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
TR2	T2CON.2	Start/stop control for Timer 2. A logic 1 starts the timer.
C/$\overline{T2}$	T2CON.1	Timer or counter select. (Timer 2) 0 = Internal timer (OSC/12) 1 = External event counter (falling edge triggered).
CP/$\overline{RL2}$	T2CON.0	Capture/Reload flag. When set, captures will occur on negative transitions at T2EX if EXEN2 = 1. When cleared, auto-reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

SU00065

Figure 1. Timer/Counter 2 (T2CON) Control Register

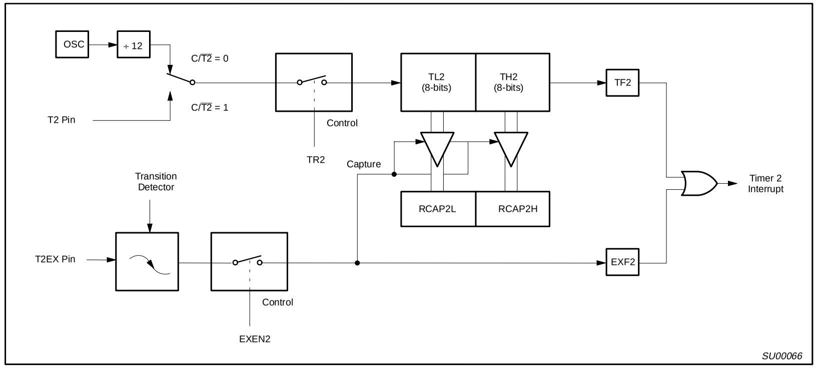


Figure 2. Timer 2 in Capture Mode

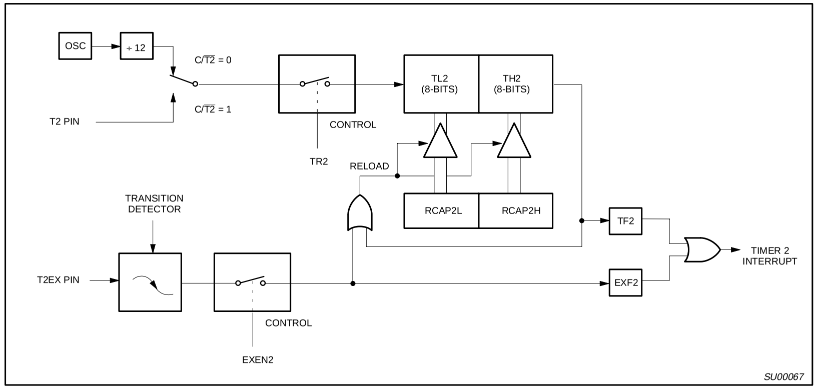


Figure 3. Timer 2 in Auto-Reload Mode

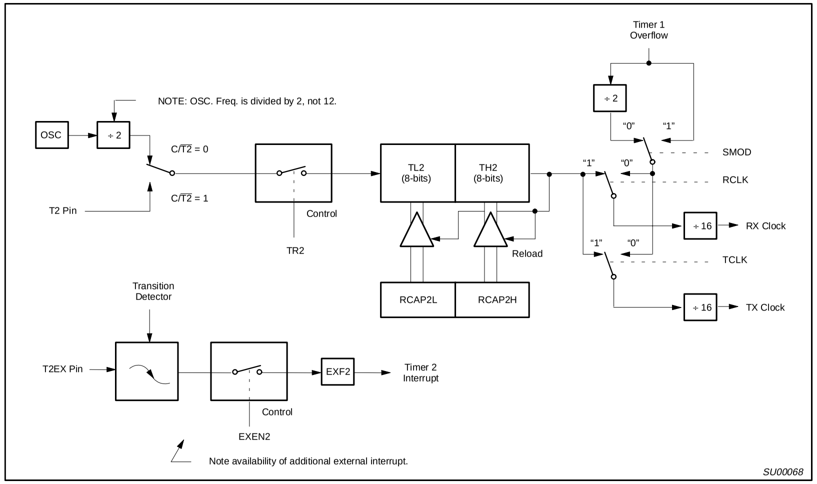


Figure 4. Timer 2 in Baud Rate Generator Mode

Table 2. Timer 2 Operating Modes

RCLK + TCLK	CP/RL2	TR2	MODE
0	0	1	16-bit Auto-reload
0	1	1	16-bit Capture
1	X	1	Baud rate generator
X	X	0	(off)

Timer 2 as a baud rate generator is shown in Figure 4. This figure is valid only if RCLK + TCLK = 1 in T2CON. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Therefore, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. Note too, that if EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt, if desired.

It should be noted that when Timer 2 is running (TR2 = 1) in "timer" function in the baud rate generator mode, one should not try to read or write TH2 or TL2. Under these conditions the timer is being incremented every state time, and the results of a read or write may not be accurate. The RCAP registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. Turn the timer off (clear TR2) before accessing the Timer 2 or RCAP registers, in this case.

Timer/Counter 2 Set-up

Except for the baud rate generator mode, the values given for T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set, separately, to turn the timer on. See Table 3 for set-up of timer 2 as a timer. See Table 4 for set-up of timer 2 as a counter.

Using Timer/Counter 2 to Generate Baud Rates

For this purpose, Timer 2 must be used in the baud rate generating mode. If Timer 2 is being clocked through pin T2 (P1.0) the baud rate is:

$$\text{Baud Rate} \quad \frac{\text{Timer 2 Overflow Rate}}{16}$$

And if it is being clocked internally, the baud rate is:

$$\text{Baud Rate} \quad \frac{\text{Oscillator Frequency}}{32 \quad [65536 \quad (\text{RCAP2H, RCAP2L})]}$$

To obtain the reload value for RCAP2H and RCA02L, the above equation can be rewritten as:

$$\text{RCAP2H, RCAP2L} \quad 65536 \quad \frac{\text{Oscillator Frequency}}{32 \quad \text{Baud Rate}}$$

Interrupts

The 80C32/87C52 has 6 interrupt sources. All except TF2 and EXF2 are identical sources to those in the 80C51.

The Interrupt Enable Register and the Interrupt Priority Register are modified to include the additional 80C32/87C52 interrupt sources. The operation of these registers is identical to the 80C51.

In the 80C32/87C52, the Timer 2 Interrupt is generated by the logical OR of TF2 and EXF2. Neither of these flags is cleared by hardware when the service routine is vectored to. In fact, the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt, and the bit will have to be cleared in software.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though it has been set or cleared by hardware. That is, interrupts can be generated or pending interrupts can be canceled in software.

The interrupt vector addresses and the interrupt priority for requests in the same priority level are given in the following:

Source	Vector Address	Priority Within Level
1. IE0	0003H	(highest)
2. TF0	000BH	
3. IE1	0013H	
4. TF1	001BH	
5. RI + TI	0023H	
6. TF2 + EXF2	002BH	(lowest)

Note that they are identical to those in the 80C51 except for the addition of the Timer 2 (TF1 and EXF2) interrupt at 002BH and at the lowest priority within a level.

Table 3. Timer 2 as a Timer

MODE	T2CON	
	INTERNAL CONTROL (Note 1)	**EXTERNAL CONTROL (Note 2)**
16-bit Auto-Reload	00H	08H
16-bit Capture	01H	09H
Baud rate generator receive and transmit same baud rate	34H	36H
Receive only	24H	26H
Transmit only	14H	16H

Table 4. Timer 2 as a Counter

MODE	TMOD	
	INTERNAL CONTROL (Note 1)	**EXTERNAL CONTROL (Note 2)**
16-bit	02H	0AH
Auto-Reload	03H	0BH

NOTES:
1. Capture/reload occurs only on timer/counter overflow.
2. Capture/reload occurs on timer/counter overflow and a 1-to-0 transition on T2EX (P1.1) pin except when timer 2 is used in the baud rate generator mode.

OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the Logic Symbol, page 4.

To drive the device from an external clock source, XTAL1 should be driven while XTAL2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles.

IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode. the control bits for the reduced power modes are in the special function register PCON.

DESIGN CONSIDERATIONS

At power-on, the voltage on V_{CC} and RST must come up at the same time for a proper start-up.

Table 5 shows the state of I/O ports during low current operating modes.

As a precaution to coming out of an unexpected power down, $\overline{INT0}$ and $\overline{INT1}$ should be disabled prior to enterring power down.

Table 5. External Pin Status During Idle and Power-Down Modes

MODE	PROGRAM MEMORY	ALE	$\overline{PSEN}$	PORT 0	PORT 1	PORT 2	PORT 3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

DALLAS SEMICONDUCTOR

DS80C320
High–Speed Micro

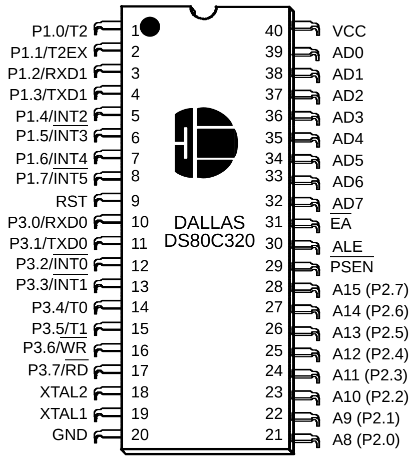
FEATURES

- 80C32–Compatible
  - Pin–compatible
  - Standard 8051 instruction set
  - Four 8–bit I/O ports
  - Three 16–bit timer/counters
  - 256 bytes scratchpad RAM
  - Multiplexed address/data bus
  - Addresses 64KB ROM and 64KB RAM
- High–speed architecture
  - 4 clocks/machine cycle (8032=12)
  - Wasted cycles removed
  - Runs DC to 33 MHz clock rates
  - Single–cycle instruction in 121 ns
  - Uses less power for equivalent work
  - Dual data pointer
  - Optional variable length MOVX to access fast/ slow RAM /peripherals
- High integration controller includes:
  - Power–fail reset
  - Programmable Watchdog timer
  - Early–warning power–fail interrupt
- Two full–duplex hardware serial ports
- 13 total interrupt sources with six external
- Available in 40–pin DIP, 44–pin PLCC and TQFP

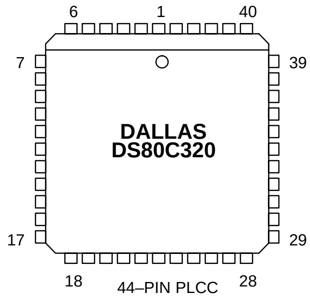
DESCRIPTION

The DS80C320 is a fast 80C31/80C32–compatible microcontroller. Wasted clock and memory cycles have been removed using a redesigned processor core. As a result, every 8051 instruction is executed between 1.5 and 3 times faster than the original for the same crystal speed. Typical applications will see a speed improvement of 2.5 times using the same code and same crystal. The DS80C320 offers a maximum crystal rate of 33 MHz, resulting in apparent execution speeds of 82.5 MHz (approximately 2.5X).

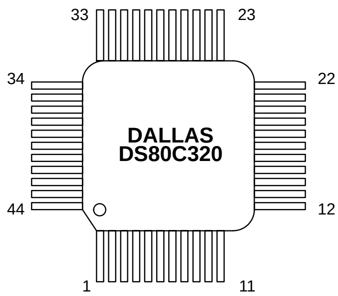
PIN ASSIGNMENT



40– PIN DIP



44–PIN PLCC



44–PIN TQFP

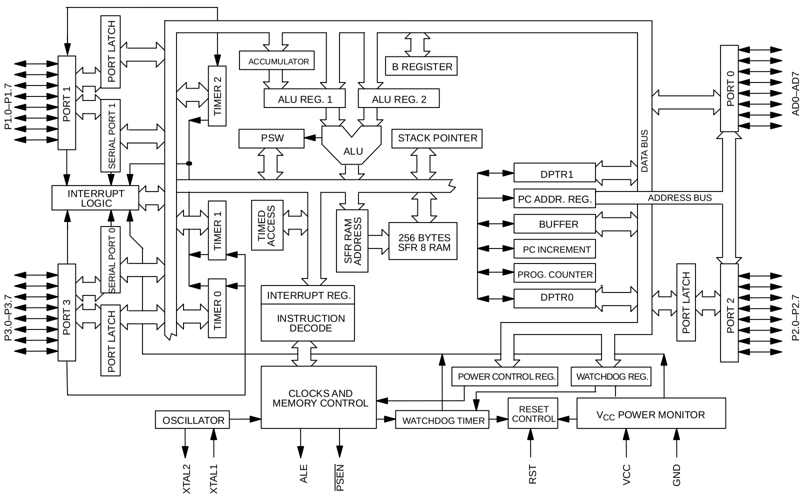
The DS80C320 is pin compatible with all three packages of the standard 80C32 and offers the same timer/counters, serial port, and I/O ports. In short, the DS80C320 is extremely familiar to 8051 users but provides the speed of a 16–bit processor.

The DS80C320 provides several extras in addition to greater speed. These include a second full hardware serial port, seven additional interrupts, programmable watchdog timer, power–fail interrupt and reset. The DS80C320 also provides dual data pointers (DPTRs) to speed block data memory moves. It can also adjust the speed of off–chip data memory access to between two and nine machine cycles for flexibility in selecting memory and peripherals.

ORDERING INFORMATION

PART NUMBER	PACKAGE	MAX CLOCK SPEED	TEMPERATURE RANGE
DS80C320–MCG	40–pin plastic DIP	25 MHz	0°C to +70°C
DS80C320–QCG	44–pin PLCC	25 MHz	0°C to +70°C
DS80C320–ECG	44–pin TQFP	25 MHz	0°C to +70°C
DS80C320–MNG	40–pin plastic DIP	25 MHz	–40°C to +85°C
DS80C320–QNG	44–pin PLCC	25 MHz	–40°C to +85°C
DS80C320–ENG	44–pin TQFP	25 MHz	–40°C to +85°C
DS80C320–MCL	40–pin plastic DIP	33 MHz	0°C to +70°C
DS80C320–QCL	44–pin PLCC	33 MHz	0°C to +70°C
DS80C320–ECL	44–pin TQFP	33 MHz	0°C to +70°C
DS80C320–MNL	40–pin plastic DIP	33 MHz	–40°C to +85°C
DS80C320–QNL	44–pin PLCC	33 MHz	–40°C to +85°C
DS80C320–ENL	44–pin TQFP	33 MHz	–40°C to +85°C

DS80C320 BLOCK DIAGRAM Figure 1



HIGH–SPEED OPERATION

The DS80C320 is built around a high speed 80C32 compatible core. Higher speed comes not just from increasing the clock frequency, but from a newer, more efficient design.

In this updated core, dummy memory cycles have been eliminated. In a conventional 80C32, machine cycles are generated by dividing the clock frequency by 12. In the DS80C320, the same machine cycle is performed in 4 clocks. Thus the fastest instruction, 1 machine cycle, is executed three times faster for the same crystal frequency. Note that these are identical instructions. A comparison of the timing differences is shown in Figure 2. The majority of instructions on the DS80C320 will see the full 3 to 1 speed improvement. Some instructions will get between 1.5 and 2.4 X improvement. Note that all instructions are faster than the original 80C51. Table 2 below shows a summary of the instruction set including the speed.

The numerical average of all opcodes is approximately a 2.5 to 1 speed improvement. Individual programs will be affected differently, depending on the actual instructions used. Speed sensitive applications would make the most use of instructions that are three times faster. However, the sheer number of 3 to 1 improved opcodes makes dramatic speed improvements likely for any code. When these architecture improvements are combined with 0.8 µm CMOS, the result is a single cycle instruction execution in 160 ns. The Dual Data Pointer feature also allows the user to eliminate wasted instructions when moving blocks of memory.

INSTRUCTION SET SUMMARY

All instructions in the DS80C320 perform the same functions as their 80C32 counterparts. Their affect on bits, flags, and other status functions is identical. However, the timing of each instruction is different. This applies both in absolute and relative number of clocks.

For absolute timing of real–time events, the timing of software loops will need to be calculated using the table below. However, counter/timers default to run at the older 12 clocks per increment. Therefore, while software runs at higher speed, timer–based events need no modification to operate as before. Timers can be set to run at 4 clocks per increment cycle to take advantage of higher speed operation.

The relative time of two instructions might be different in the new architecture than it was previously. For example, in the original architecture, the "MOVX A, @DPTR" instruction and the "MOV direct, direct" instruction used two machine cycles or 24 oscillator cycles. Therefore, they required the same amount of time. In the DS80C320, the MOVX instruction can be done in two machine cycles or eight oscillator cycles but the "MOV direct, direct" uses three machine cycles or 12 oscillator cycles. While both are faster than their original counterparts, they now have different execution times from each other. This is because in most cases, the DS80C320 uses one cycle for each byte. The user concerned with precise program timing should examine the timing of each instruction for familiarity with the changes. Note that a machine cycle now requires just four clocks, and provides one ALE pulse per cycle. Many instructions require only one cycle, but some require five. In the original architecture, all were one or two cycles except for MUL and DIV.

INSTRUCTION SET SUMMARY Table 2

Legends:

A	–	Accumulator
Rn	–	Register R7–R0
direct	–	Internal Register address
@Ri	–	Internal Register pointed–to by R0 or R1 (except MOVX)
rel	–	2's complement offset byte
bit	–	direct bit–address
#data	–	8–bit constant
#data 16	–	16–bit constant
addr 16	–	16–bit destination address
addr 11	–	11–bit destination address

INSTRUCTION	BYTE	OSCILLATOR CYCLES	INSTRUCTION	BYTE	OSCILLATOR CYCLES
Arithmetic Instructions:					
ADD A, Rn	1	4	INC A	1	4
ADD A, direct	2	8	INC Rn	1	4
ADD A, @Ri	1	4	INC direct	2	8
ADD A, #data	2	8	INC @Ri	1	4
ADDC A, Rn	1	4	INC DPTR	1	12
ADDC A, direct	2	8	DEC A	1	4
ADDC A, @Ri	1	4	DEC Rn	1	4
ADDC A, #data	2	8	DEC direct	2	8
SUBB A, Rn	1	4	DEC @Ri	1	4
SUBB A, direct	2	8	MUL AB	1	20
SUBB A, @Ri	1	4	DIV AB	1	20
SUBB A, #data	2	8	DA A	1	4
Logical Instructions:					
ANL A, Rn	1	4	XRL A, Rn	1	4
ANL A, direct	2	8	XRL A, direct	2	8
ANL A, @Ri	1	4	XRL A, @Ri	1	4
ANL A, #data	2	8	XRL A, #data	2	8
ANL direct, A	2	8	XRL direct, A	2	8
ANL direct, #data	3	12	XRL direct, #data	3	12
ORL A, Rn	1	4	CLR A	1	4
ORL A, direct	2	8	CPL A	1	4
ORL A, @Ri	1	4	RL A	1	4
ORL A, #data	2	8	RLC A	1	4
ORL direct, A	2	8	RR A	1	4
ORL direct, #data	3	12	RRC A	1	4
			SWAP A	1	4
Data Transfer Instructions:					
MOV A, Rn	1	4	MOVC A, @A+DPTR	1	12
MOV A, direct	2	8	MOVC A, @A+PC	1	12
MOV A, @Ri	1	4	MOVX A, @Ri	1	8–36 *
MOV A, #data	2	8	MOVX A, @DPTR	1	8–36 *
MOV Rn, A	1	4	MOVX @Ri, A	1	8–36 *
MOV Rn, direct	2	8	MOVX @DPTR, A	1	8–36 *
MOV Rn, #data	2	8	PUSH direct	2	8
MOV direct, A	2	8	POP direct	2	8
MOV direct, Rn	2	8	XCH A, Rn	1	4
MOV direct1, direct2	3	12	XCH A, direct	2	8
MOV direct, @Ri	2	8	XCH A, @Ri	1	4
MOV direct, #data	3	12	XCHD A, @Ri	1	4
MOV @Ri, A	1	4			
MOV @Ri, direct	2	8			
MOV @Ri, #data	2	8			
MOV DPTR, #data 16	3	12			

*User Selectable

Bit Manipulation Instructions:					
CLR C	1	4	ANL C, bit	2	8
CLR bit	2	8	ANL C, $\overline{\text{bit}}$	2	8
SETB C	1	4	ORL C, bit	2	8
SETB bit	2	8	ORL C, $\overline{\text{bit}}$	2	8
CPL C	1	4	MOV C, bit	2	8
CPL bit	2	8	MOV bit, C	2	8
Program Branching Instructions:					
ACALL addr 11	2	12	CJNE A, direct, rel	3	16
LCALL addr 16	3	16	CJNE A, #data, rel	3	16
RET	1	16	CJNE Rn, #data, rel	3	16
RETI	1	16	CJNE Ri, #data, rel	3	16
AJMP addr 11	2	12	NOP	1	4
LJMP addr 16	3	16	JC rel	2	12
SJMP rel	2	12	JNC rel	2	12
JMP @A+DPTR	1	12	JB bit, rel	3	16
JZ rel	2	12	JNB bit, rel	3	16
JNZ rel	2	12	JBC bit, rel	3	16
DJNZ Rn, rel	2	12			
DJNZ direct, rel	3	16			

The table above shows the speed for each class of instruction. Note that many of the instructions have multiple opcodes. There are 255 opcodes for 111 instructions. Of the 255 opcodes, 159 are three times faster than the original 80C32. While a system that emphasizes those instructions will see the most improvement, the large total number that receive a 3 to 1 improvement assure a dramatic speed increase for any system. The speed improvement summary is provided below.

SPEED ADVANTAGE SUMMARY

#Opcodes	Speed Improvement
159	3.0 x
51	1.5 x
43	2.0 x
2	2.4 x
255	Average: 2.5

MEMORY ACCESS

The DS80C320 contains no on–chip ROM and 256 bytes of scratchpad RAM. Off–chip memory is accessed using the multiplexed address/data bus on P0 and the MSB address on P2. A typical memory connection is shown in Figure 3. Timing diagrams are provided in the Electrical Specifications. Program memory (ROM) is accessed at a fixed rate determined by the crystal frequency and the actual instructions. As mentioned above, an instruction cycle requires four clocks. Data memory (RAM) is accessed according to a variable speed MOVX instruction as described below.

DATA MEMORY CYCLE STRETCH VALUES Table 3

CKCON.2–0 MD2	MD1	MD0	MEMORY CYCLES	RD or WR STROBE WIDTH IN CLOCKS	STROBE WIDTH TIME @ 25 MHz
0	0	0	2	2	80 ns
0	0	1	3 (default)	4	160 ns
0	1	0	4	8	320 ns
0	1	1	5	12	480 ns
1	0	0	6	16	640 ns
1	0	1	7	20	800 ns
1	1	0	8	24	960 ns
1	1	1	9	28	1120 ns

DUAL DATA POINTER

Data memory block moves can be accelerated using the DS80C320 Dual Data Pointer (DPTR). The standard 8032 DPTR is a 16–bit value that is used to address off–chip data RAM or peripherals. In the DS80C320, the standard data pointer is called DPTR 0 and is located at SFR addresses 82h and 83h. These are the standard locations. No modification of standard code is needed to use DPTR. The new DPTR is located at SFR 84h and 85h and is called DPTR1. The DPTR Select bit (DPS) chooses the active pointer and is located at the LSB of the SFR location 86h. No other bits in register 86h have any effect and are set to 0. The user switches between data pointers by toggling the LSB of register 86h. The increment (INC) instruction is the fastest way to accomplish this. All DPTR–related instructions use the currently selected DPTR for any activity. Therefore only one instruction is required to switch from a source to a destination address. Using the Dual–Data Pointer saves code from needing to save source and destination addresses when doing a block move. Once loaded, the software simply switches between DPTR and 1. The relevant register locations are as follows.

DPL	82h	Low byte original DPTR
DPH	83h	High byte original DPTR
DPL1	84h	Low byte new DPTR
DPH1	85h	High byte new DPTR
DPS	86h	DPTR Select (LSB)

Sample code listed below illustrates the saving from using the dual DPTR. The example program was original code written for an 8051 and requires a total of 1869 machine cycles on the DS80C320. This takes 299 µs to execute at 25 MHz. The new code using the Dual DPTR requires only 1097 machine cycles taking 175.5 µs. The Dual DPTR saves 772 machine cycles or 123.5 µs for a 64 byte block move. Since each pass through the loop saves 12 machine cycles when compared to the single DPTR approach, larger blocks gain more efficiency using this feature.

64 BYTE BLOCK MOVE WITHOUT DUAL DATA POINTER

```
; SH and SL are high and low byte source address.
; DH and DL are high and low byte of destination address.
                                                        # CYCLES

MOV     R5, #64d       ; NUMBER OF BYTES TO MOVE              2
MOV     DPTR, #SHSL    ; LOAD SOURCE ADDRESS                  3
MOV     R1, #SL        ; SAVE LOW BYTE OF SOURCE              2
MOV     R2, #SH        ; SAVE HIGH BYTE OF SOURCE             2
MOV     R3, #DL        ; SAVE LOW BYTE OF DESTINATION         2
MOV     R4, #DH        ; SAVE HIGH BYTE OF DESTINATION        2

MOVE:
; THIS LOOP IS PERFORMED THE NUMBER OF TIMES LOADED INTO R5, IN THIS EXAMPLE 64

MOVX    A, @DPTR       ; READ SOURCE DATA BYTE                2
MOV     R1, DPL        ; SAVE NEW SOURCE POINTER              2
MOV     R2, DPH        ;                                      2
```



```
MOV     DPL, R3        ; LOAD NEW DESTINATION                 2
MOV     DPH, R4        ;                                      2
MOVX    @DPTR, A       ; WRITE DATA TO DESTINATION            2
INC     DPTR           ; NEXT DESTINATION ADDRESS             3
MOV     R3, DPL        ; SAVE NEW DESTINATION POINTER         2
MOV     R4, DPH        ;                                      2
MOV     DPL, R1        ; GET NEW SOURCE POINTER               2
MOV     DPH, R2        ;                                      2
INC     DPTR           ; NEXT SOURCE ADDRESS                  3
DJNZ    R5, MOVE       ; FINISHED WITH TABLE?                 3
```

64 BYTE BLOCK MOVE WITH DUAL DATA POINTER

```
; SH and SL are high and low byte source address.
; DH and DL are high and low byte of destination address.
; DPS is the data pointer select. Reset condition is DPS=0, DPTR0 is selected.
                                                        # CYCLES

EQU     DPS, #86h      ; TELL ASSEMBLER ABOUT DPS

MOV     R5, #64        ; NUMBER OF BYTES TO MOVE              2
MOV     DPTR, #DHDL    ; LOAD DESTINATION ADDRESS             3
INC     DPS            ; CHANGE ACTIVE DPTR                   2
MOV     DPTR, #SHSL    ; LOAD SOURCE ADDRESS                  2

MOVE:
; THIS LOOP IS PERFORMED THE NUMBER OF TIMES LOADED INTO R5, IN THIS EXAMPLE 64

MOVX    A, @DPTR       ; READ SOURCE DATA BYTE                2
INC     DPS            ; CHANGE DPTR TO DESTINATION           2
MOVX    @DPTR, A       ; WRITE DATA TO DESTINATION            2
INC     DPTR           ; NEXT DESTINATION ADDRESS             3
INC     DPS            ; CHANGE DATA POINTER TO SOURCE        2
INC     DPTR           ; NEXT SOURCE ADDRESS                  3
DJNZ    R5, MOVE       ; FINISHED WITH TABLE?                 3
```

PERIPHERAL OVERVIEW

Peripherals in the DS80C320 are accessed using Special Function Registers (SFRs). The DS80C320 provides several of the most commonly needed peripheral functions in microcomputer–based systems. These functions are new to the 80C32 family and include a second serial port, Power–fail Reset, Power–fail Interrupt, and a programmable Watchdog Timer. These are described below, and more details are available in the High–Speed Microcontroller User's Guide.

SERIAL PORTS

The DS80C320 provides a serial port (UART) that is identical to the 80C32. Many applications require serial communication with multiple devices. Therefore the DS80C320 provides a second hardware serial port that is a full duplicate of the standard one. It optionally uses pins P1.2 (RXD1) and P1.3 (TXD1). This port has duplicate control functions included in new SFR locations.

The second serial port operates in a comparable manner with the first. Both can operate simultaneously but can be at different baud rates.

The second serial port has similar control registers (SCON1 at C0h, SBUF1 at C1h) to the original. One difference is that for timer based baud rates, the original serial port can use Timer 1 or Timer 2 to generate baud rates. This is selected via SFR bits. The new serial port can only use Timer 1.

TIMER RATE CONTROL

One important difference exists between the DS80C320 and 80C32 regarding timers. The original 80C32 used a 12 clock per cycle scheme for timers and consequently for some serial baud rates (depending on the mode). The DS80C320 architecture normally runs using 4 clocks per cycle. However, in the area of timers, the DS80C320 will default to a 12 clock per cycle

scheme on a reset. This allows existing code with real–time dependencies such as baud rates to operate properly. If an application needs higher speed timers or serial baud rates, the timers can be set to run at the 4 clock rate.

The Clock Control register (CKCON – 8Eh) determines these timer speeds. When the relevant CKCON bit is a logic 1, the DS80C320 uses 4 clocks per cycle to generate timer speeds. When the control bit is set to a 0, the DS80C320 uses 12 clocks for timer speeds. The reset condition is a 0. CKCON.5 selects the speed of Timer 2. CKCON.4 selects Timer 1 and CKCON.3 selects Timer 0. Note that unless a user desires very fast timing, it is unnecessary to alter these bits. Note that the timer controls are independent.

POWER FAIL RESET

The DS80C320 incorporates a precision band–gap voltage reference to determine when V_{CC} is out–of–tolerance. While powering up, internal circuits will hold the DS80C320 in a reset state until V_{CC} rises above the V_{RST} reset threshold. Once V_{CC} is above this level, the oscillator will begin running. An internal reset circuit will then count 65536 clocks to allow time for power and the oscillator to stabilize. The microcontroller will then exit the reset condition. No external components are needed to generate a power on reset. During power down or during a severe power glitch, as V_{CC} falls below V_{RST}, the microcontroller will also generate its own reset. It will hold the reset condition as long as power remains below the threshold. This reset will occur automatically, needing no action from the user or from the software. Refer to the Electrical Specifications for the exact value of V_{RST}.

POWER FAIL INTERRUPT

The same reference that generates a precision reset threshold can also generate an optional early warning Power–fail Interrupt (PFI). When enabled by the application software, this interrupt always has the highest priority. On detecting that the V_{CC} has dropped below V_{PFW} and that the PFI is enabled, the processor will vector to ROM address 0033h. The PFI enable is located in the Watchdog Control SFR (WDCON – D8h). Setting WDCON.5 to a logic one will enable the PFI. The application software can also read a flag at WDCON.4. This bit is set when a PFI condition has occurred. The flag is independent of the interrupt enable and software must manually clear it.

WATCHDOG TIMER

For applications that can not afford to run out–of–control, the DS80C320 incorporates a programmable Watchdog Timer circuit. It resets the uC if software fails to reset the Watchdog before the selected time interval has elapsed. The user selects one of four time–out values. After enabling the Watchdog, software must reset the timer prior to expiration of the interval, or the CPU will be reset. Both the Watchdog Enable and the Watchdog Reset bits are protected by a "Timed Access" circuit. This prevents accidentally clearing the Watchdog. Time–out values are precise since they are related to the crystal frequency as shown below in Table 4. For reference, the time periods at 25 MHz are also shown.

The DS80C320 Watchdog also provides a useful option for systems that may not require a reset. If enabled, then 512 clocks before giving a reset, the Watchdog will give an interrupt. The interrupt can also serve as a convenient time–base generator, or be used to wake–up the processor from Idle mode. The Watchdog function is controlled in the Clock Control (CKCON – 8Eh), Watchdog Control (WDCON – D8h), and Extended Interrupt Enable (EIE – E8h) SFRs. CKCON.7 and CKCON.6 are called WD1 and WD0 respectively and are used to select the Watchdog time–out period as shown in Table 4.

WATCHDOG TIME–OUT VALUES Table 4

WD1	WD0	INTERRUPT TIME–OUT	TIME (@25 MHz)	RESET TIME–OUT	TIME (@25 MHz)
0	0	2^{17} clocks	5.243 ms	2^{17} + 512 clocks	5.263 ms
0	1	2^{20} clocks	41.94 ms	2^{20} + 512 clocks	41.96 ms
1	0	2^{23} clocks	335.54 ms	2^{23} + 512 clocks	335.56 ms
1	1	2^{26} clocks	2684.35 ms	2^{26} + 512 clocks	2684.38 ms

As shown above, the Watchdog Timer uses the crystal frequency as a time base. A user selects one of four counter values to determine the time–out. These clock counter lengths are 2^{17}= 131,072 clocks; 2^{20} = 1,048,576; 2^{23} = 8,388,608 clocks; or 2^{26} = 67,108,864 clocks. The times shown in Table 4 above are with a 25 MHz crystal frequency. Note that once the counter chain has reached a conclusion, the optional interrupt is generated. Regardless of whether the user enables this interrupt, there are then 512 clocks left until a reset occurs. There are five control bits in special function registers that affect the Watchdog Timer and two status flags that report to the user.

WDIF (WDCON.3) is the interrupt flag that is set when there are 512 clocks remaining until a reset occurs. WTRF (WDCON.2) is the flag that is set when a Watchdog reset has occurred. This allows the application software to determine the source of a reset.

EWT (WDCON.1) is the enable for the Watchdog Timer. Software sets this bit to enable the timer. The bit is protected by Timed Access discussed below. RWT (WDCON.0) is the bit that software uses to restart the Watchdog Timer. Setting this bit restarts the timer for another full interval. Application software must set this bit prior to the time–out. As mentioned previously, WD1 and 0 (CKCON .7 and 6) select the time–out. Finally, the Watchdog Interrupt is enabled using EWDI (EIE.4). The Special Function Register map is shown below.

INTERRUPTS

The DS80C320 provides 13 sources of interrupt with three priority levels. The Power–fail Interrupt (PFI), if enabled, always has the highest priority. There are two remaining user selectable priorities: high and low. If two interrupts that have the same priority occur simultaneously, the natural precedence given below determines which is a acted upon. Except for the PFI, all interrupts that are new to the 8051 family have a lower natural priority than the originals.

INTERRUPT PRIORITY Table 5

NAME	DESCRIPTION	VECTOR	NATURAL PRIORITY	OLD/NEW
PFI	Power Fail Interrupt	33h	1	NEW
$\overline{INT0}$	External Interrupt 0	03h	2	OLD
TF0	Timer 0	0Bh	3	OLD
$\overline{INT1}$	External Interrupt 1	13h	4	OLD
TF1	Timer 1	1Bh	5	OLD
SCON0	TI0 or RI0 from serial port 0	23h	6	OLD
TF2	Timer 2	2Bh	7	OLD
SCON1	TI1 or RI1 from serial port 1	3Bh	8	NEW
$\overline{INT2}$	External Interrupt 2	43h	9	NEW
$\overline{INT3}$	External Interrupt 3	4Bh	10	NEW
INT4	External Interrupt 4	53h	11	NEW
$\overline{INT5}$	External Interrupt 5	5Bh	12	NEW
WDTI	Watchdog Time–out Interrupt	63h	13	NEW

POWER MANAGEMENT

The DS80C320 provides the standard Idle and power down (Stop) that are available on the standard 80C32. However the DS80C320 has enhancements that make these modes more useful, and allow more power saving.

The Idle mode is invoked by setting the LSB of the Power Control register (PCON – 87h). Idle will leave internal clocks, serial port and timer running. No memory access will be performed so power is dramatically reduced. Since clocks are running, the Idle power consumption is related to crystal frequency. It should be approximately 1/2 of the operational power. The CPU can exit the Idle state with any interrupt or a reset.

The power–down or Stop mode is invoked by setting the PCON.1 bit. Stop mode is a lower power state than Idle since it turns off all internal clocking. The I_{CC} of a standard Stop mode is approximately 1 µA but is specified in the Electrical Specifications. The CPU will exit Stop mode from an external interrupt or a reset condition.

Note that internally generated interrupts (timer, serial port, watchdog) are not useful since they require clocking activity.

IDLE MODE ENHANCEMENTS

A simple enhancement to Idle mode makes it substantially more useful. The innovation involves not the Idle mode itself, but the watchdog timer. As mentioned above, the Watchdog Timer provides an optional interrupt capability. This interrupt can provide a periodic interval timer to bring the DS80C320 out of Idle mode. This can be useful even if the Watchdog is not normally used. By enabling the Watchdog Timer and its interrupt prior to invoking Idle, a user can periodically come out of Idle perform an operation, then return to Idle until the next operation. This will lower the overall power consumption. When using the Watchdog Interrupt to cancel the Idle state, make sure to restart the Watchdog Timer or it will cause a reset.

STOP MODE ENHANCEMENTS

The DS80C320 provides two enhancements to the Stop mode. As documented above, the DS80C320 provides a band–gap reference to determine Power–fail Interrupt and Reset thresholds. The default state is that the band–gap reference is off when Stop mode is invoked. This allows the extremely low power state mentioned above. A user can optionally choose to have the band–gap enabled during Stop mode. This means that PFI and power–fail reset will be activated and are valid means for leaving Stop mode.

In Stop mode with the band–gap on, I_{CC} will be approximately 50 µA compared with 1 µA with the band–gap off. If a user does not require a Power–fail Reset or Interrupt while in Stop mode, the band–gap can remain turned off. Note that only the most power sensitive applications should turn off the band–gap, as this results in an uncontrolled power down condition.

The control of the band–gap reference is located in the Extended Interrupt Flag register (EXIF – 91h). Setting BGS (EXIF.0) to a one will leave the band–gap reference enabled during Stop mode. The default or reset condition is with the bit at a logic 0. This results in the band–gap being turned off during Stop mode. Note that this bit has no control of the reference during full power or Idle modes.

The second feature allows an additional power saving option. This is the ability to start instantly when exiting Stop mode. It is accomplished using an internal ring oscillator that can be used when exiting Stop mode in response to an interrupt. The benefit of the ring oscillator is as follows.

Using Stop mode turns off the crystal oscillator and all internal clocks to save power. This requires that the oscillator be restarted when exiting Stop mode. Actual start–up time is crystal dependent, but is normally at least 4 ms. A common recommendation is 10 ms. In an application that will wake–up, perform a short operation, then return to sleep, the crystal start–up can be longer than the real transaction. However, the ring oscillator will start instantly. The user can perform a simple operation and return to sleep before the crystal has even stabilized. If the ring is used to start and the processor remains running, hardware will automatically switch to the crystal once a power–on reset interval (65536 clocks) has expired. This value is used to guarantee stability even though power is not being cycled.

If the user returns to Stop mode prior to switching of crystal, then all clocks will be turned off again. The ring oscillator runs at approximately 4 MHz but will not be a precision value. No real–time precision operations (including serial communication) should be conducted during this ring period. Figure 7 shows how the operation would compare when using the ring, and when starting up normally. The default state is to come out of Stop mode without using the ring oscillator.

This function is controlled using the RGSL – Ring Select bit at EXIF.1 (EXIF – 91h). When EXIF.1 is set, the ring oscillator will be used to come out of Stop mode quickly. As mentioned above, the processor will automatically switch from the ring (if enabled) to the crystal after a delay of 65536 crystal clocks. For a 3.57 MHz crystal, this is approximately 18 ms. The processor sets a flag called RGMD – Ring Mode to tell software that the ring is being used. This bit at EXIF.2 will be a logic 1 when the ring is in use. No serial communication or precision timing should be attempted while this bit is set, since the operating frequency is not precise.

RING OSCILLATOR START–UP Figure 4

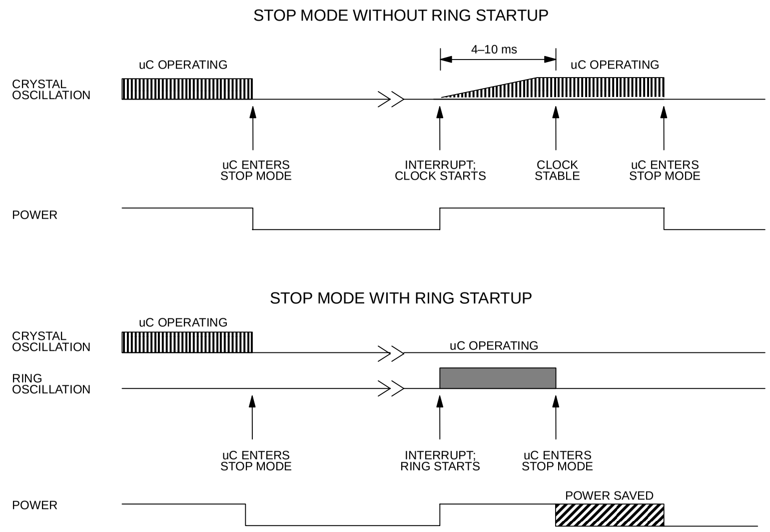


Diagram assumes that the operation following Stop requires less than 18 ms complete.

TIMED ACCESS PROTECTION

Selected SFR bits are critical to operation, making it desirable to protect against an accidental write operation. The Timed Access procedure prevents an errant cpu from accidentally altering a bit that would cause difficulty. The Timed Access procedure requires that the write of a protected bit be preceded by the following instructions :

```
MOV    0C7h, #0AAh
MOV    0C7h, #55h
```

By writing an AAh followed by a 55h to the Timed Access register (location C7h), the hardware opens a two cycle window that allows software to modify one of the protected bits. If the instruction that seeks to modify the protected bit is not immediately proceeded by these instructions, the write will not take effect. The protected bits are:

EXIF.0	BGS Band–gap Select
WDCON.6	POR Power–on Reset flag
WDCON.1	EWT Enable Watchdog
WDCON.0	RWT Reset Watchdog
WDCON.3	WDIF Watchdog Interrupt Flag

SPECIAL FUNCTION REGISTERS

Most special features of the DS80C320 or 80C32 are controlled by bits in special function registers (SFRs). This allows the DS80C320 to add many features but use the same instruction set. When writing software to use a new feature, the SFR must be defined to an assembler or compiler using an equate statement. This is the only change needed to access the new function. The DS80C320 duplicates the SFRs that are contained in the standard 80C32. Table 6 shows the register addresses and bit locations. Many are standard 80C32 registers. The High–Speed Microcontroller User's Guide describes all SFRs.

DS80C320

SPECIAL FUNCTION REGISTER LOCATIONS Table 6

REGISTER	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	ADDRESS
SP									81h
DPL									82h
DPH									83h
DPL1									84h
DPH1									85h
DPS	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**SEL**	86h
PCON	SMOD_0	**SMOD0**	–	–	GF1	GF0	STOP	IDLE	87h
TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	88h
TMOD	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0	89h
TL0									8Ah
TL1									8Bh
TH0									8Ch
TH1									8Dh
CKCON	**WD1**	**WD0**	**T2M**	**T1M**	**T0M**	**MD2**	**MD1**	**MD0**	8Eh
P1	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	90h
EXIF	**IE5**	**IE4**	**IE3**	**IE2**	**–**	**RGMD**	**RGSL**	**BGS**	91h
SCON0	SM0/FE_0	SM1_0	SM2_0	REN_0	TB8_0	RB8_0	TI_0	RI_0	98h
SBUF0									99h
P2	P2.0	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	A0h
IE	EA	ES1	ET2	ES0	ET1	EX1	ET0	EX0	A8h
SADDR0									A9h
SADDR1									AAh
P3	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	B0h
IP	–	PS1	PT2	PS0	PT1	PX1	PT0	PX0	B8h
SADEN0									B9h
SADEN1									BAh
SCON1	**SM0/FE_0**	**SM1_0**	**SM2_0**	**REN_0**	**TB8_0**	**RB8_0**	**TI_0**	**RI_0**	C0h
SBUF1									C1h
STATUS	PIP	HIP	LIP	1	1	1	1	1	C5h
TA									C7h
T2CON	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/RL2	C8h
T2MOD	–	–	–	–	–	–	T2OE	DCEN	C9h
RCAP2L									CAh
RCAP2H									CBh
TL2									CCh
TH2									CDh
PSW	CY	AC	F0	RS1	RS0	OV	FL	P	D0h
WDCON	**SMOD_1**	**POR**	**EPFI**	**PFI**	**WDIF**	**WTRF**	**EWT**	**RWT**	D8h
ACC									E0h
EIE	**–**	**–**	**–**	**EWDI**	**EX5**	**EX4**	**EX3**	**EX2**	E8h
B									F0h
EIP	**–**	**–**	**–**	**PWDI**	**PX5**	**PX4**	**PX3**	**PX2**	F8h

052296 16/33

PROGRAMMER'S GUIDE AND INSTRUCTION SET

Memory Organization

Program Memory
The 80C51 has separate address spaces for program and data memory. The Program memory can be up to 64k bytes long. The lower 4k can reside on-chip. Figure 1 shows a map of the 80C51 program memory.

The 80C51 can address up to 64k bytes of data memory to the chip. The MOVX instruction is used to access the external data memory.

The 80C51 has 128 bytes of on-chip RAM, plus a number of Special Function Registers (SFRs). The lower 128 bytes of RAM can be accessed either by direct addressing (MOV data addr) or by indirect addressing (MOV @Ri). Figure 2 shows the Data Memory organization.

Direct and Indirect Address Area
The 128 bytes of RAM which can be accessed by both direct and indirect addressing can be divided into three segments as listed below and shown in Figure 3.

1. Register Banks 0-3: Locations 0 through 1FH (32 bytes). The device after reset defaults to register bank 0. To use the other register banks, the user must select them in software. Each register bank contains eight 1-byte registers 0 through 7. Reset initializes the stack pointer to location 07H, and it is incremented once to start from location 08H, which is the first register (R0) of the second register bank. Thus, in order to use more than one register bank, the SP should be initialized to a different location of the RAM where it is not used for data storage (i.e., the higher part of the RAM).
2. Bit Addressable Area: 16 bytes have been assigned for this segment, 20H-2FH. Each one of the 128 bits of this segment can be directly addressed (0-7FH). The bits can be referred to in two ways, both of which are acceptable by most assemblers. One way is to refer to their address (i.e., 0-7FH). The other way is with reference to bytes 20H to 2FH. Thus, bits 0-7 can also be referred to as bits 20.0-20.7, and bits 8-FH are the same as 21.0-21.7, and so on. Each of the 16 bytes in this segment can also be addressed as a byte.
3. Scratch Pad Area: 30H through 7FH are available to the user as data RAM. However, if the stack pointer has been initialized to this area, enough bytes should be left aside to prevent SP data destruction.

Figure 2 shows the different segments of the on-chip RAM.

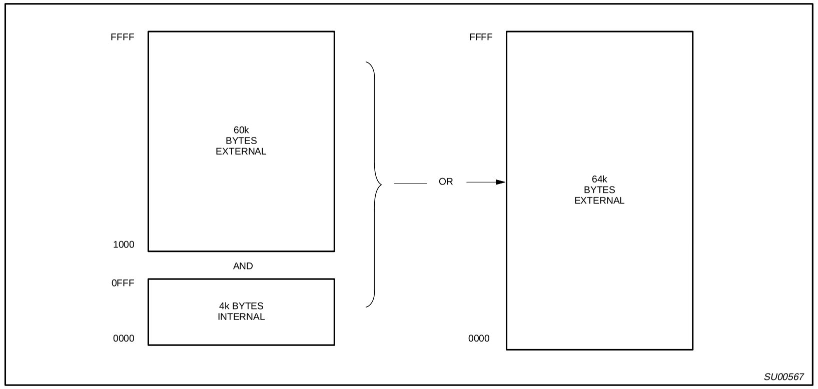


Figure 1. 80C51 Program Memory

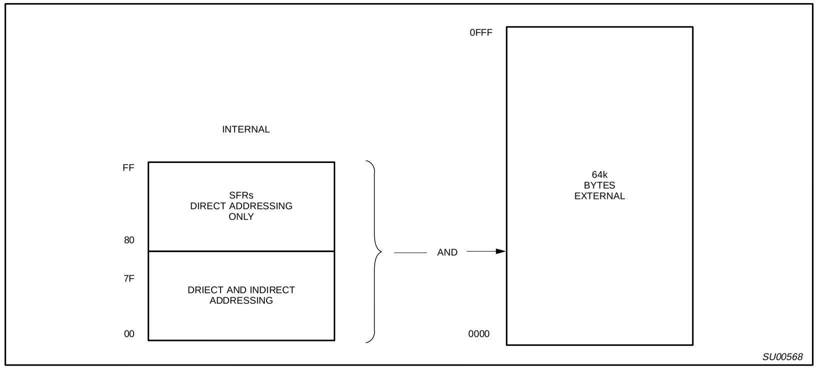


Figure 2. 80C51 Data Memory

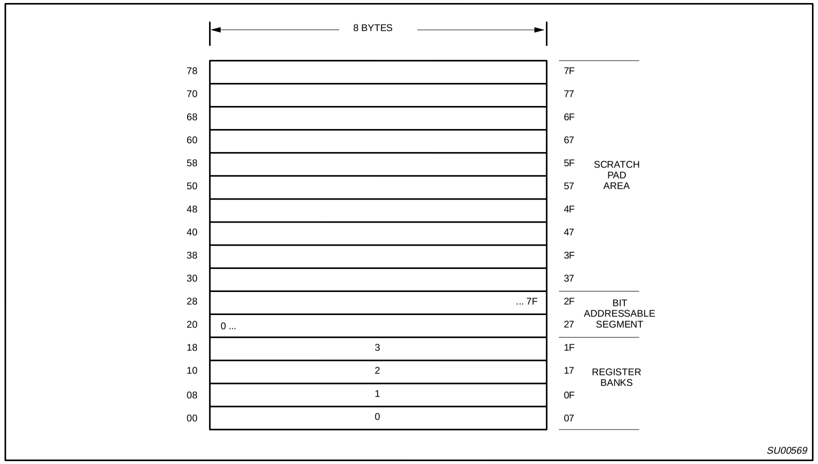


Figure 3. 128 Bytes of RAM Direct and Indirect Addressable

Table 1. 80C51 Special Function Registers

SYMBOL	DESCRIPTION	DIRECT ADDRESS	BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION MSB							LSB	RESET VALUE
ACC*	Accumulator	E0H	E7	E6	E5	E4	E3	E2	E1	E0	00H
B*	B register	F0H	F7	F6	F5	F4	F3	F2	F1	F0	00H
DPTR	Data pointer (2 bytes)										
DPH	Data pointer high	83H									00H
DPL	Data pointer low	82H									00H
			AF	AE	AD	AC	AB	AA	A9	A8	
IE*	Interrupt enable	A8H	EA	–	–	ES	ET1	EX1	ET0	EX0	0x000000B
			BF	BE	BD	BC	BB	BA	B9	B8	
IP*	Interrupt priority	B8H	–	–	–	PS	PT1	PX1	PT0	PX0	xx000000B
			87	86	85	84	83	82	81	80	
P0*	Port 0	80H	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	FFH
			97	96	95	94	93	92	91	90	
P1*	Port 1	90H	–	–	–	–	–	–	T2EX	T2	FFH
			A7	A6	A5	A4	A3	A2	A1	A0	
P2*	Port 2	A0H	A15	A14	A13	A12	A11	A10	A9	A8	FFH
			B7	B6	B5	B4	B3	B2	B1	B0	
P3*	Port 3	B0H	$\overline{RD}$	$\overline{WR}$	T1	T0	$\overline{INT1}$	$\overline{INT0}$	TxD	Rxd	FFH
PCON[1]	Power control	87H	SMOD	–	–	–	GF1	GF0	PD	IDL	0xxxxxxxB
			D7	D6	D5	D4	D3	D2	D1	D0	
PSW*	Program status word	D0H	CY	AC	F0	RS1	RS0	OV	–	P	00H
SBUF	Serial data buffer	99H									xxxxxxxxB
			9F	9E	9D	9C	9B	9A	99	98	
SCON*	Serial controller	98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00H
SP	Stack pointer	81H									07H
			8F	8E	8D	8C	8B	8A	89	88	
TCON*	Timer control	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
TH0	Timer high 0	8CH									00H
TH1	Timer high 1	8DH									00H
TL0	Timer low 0	8AH									00H
TL1	Timer low 1	8BH									00H
TMOD	Timer mode	89H	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0	00H

NOTES:
* Bit addressable
1. Bits GF1, GF0, PD, and IDL of the PCON register are not implemented on the NMOS 8051/8031.

Philips Semiconductors

80C51 Family

80C51 family programmer's guide and instruction set

8 BYTES

F8									FF
F0	B								F7
E8									EF
E0	ACC								E7
D8									DF
D0	PSW								D7
C8									CF
C0									C7
B8	IP								BF
B0	P3								B7
A8	IE								AF
A0	P2								A7
98	SCON	SBUF							9F
90	P1								97
88	TCON	TMOD	TL0	TL1	TH0	TH1			8F
80	P0	SP	DPL	DPH				PCON	87

↑ BIT ADDRESSABLE

SU00570

Figure 4. SFR Memory Map

Those SFRs that have their bits assigned for various functions are listed in this section. A brief description of each bit is provided for quick reference. For more detailed information refer to the Architecture Chapter of this book.

PSW: PROGRAM STATUS WORD. BIT ADDRESSABLE.

CY	AC	F0	RS1	RS0	OV	–	P

CY PSW.7 Carry Flag.
AC PSW.6 Auxiliary Carry Flag.
F0 PSW.5 Flag 0 available to the user for general purpose.
RS1 PSW.4 Register Bank selector bit 1 (SEE NOTE 1).
RS0 PSW.3 Register Bank selector bit 0 (SEE NOTE 1).
OV PSW.2 Overflow Flag.
– PSW.1 Usable as a general purpose flag.
P PSW.0 Parity flag. Set/cleared by hardware each instruction cycle to indicate an odd/even number of '1' bus in the accumulator.

NOTE:
1. The value presented by RS0 and RS1 selects the corresponding register bank.

RS1	RS0	REGISTER BANK	ADDRESS
0	0	0	00H-07H
0	1	1	08H-0FH
1	0	2	10H-17H
1	1	3	18H-1FH

PCON: POWER CONTROL REGISTER. NOT BIT ADDRESSABLE.

SMOD	–	–	–	GF1	GF0	PD	IDL

SMOD Double baud rate bit. If Timer 1 is used to generate baud rate and SMOD = 1, the baud rate is doubled when the Serial Port is used in modes 1, 2, or 3.
– Not implemented, reserved for future use.*
– Not implemented reserved for future use.*
– Not implemented reserved for future use.*
GF1 General purpose flag bit.
GF0 General purpose flag bit.
PD Power Down Bit. Setting this bit activates Power Down operation in the 80C51. (Available only in CMOS.)
IDL Idle mode bit. Setting this bit activates Idle Mode operation in the 80C51. (Available only in CMOS.)

If 1s are written to PD and IDL at the same time, PD takes precedence.

* User software should not write 1s to reserved bits. These bits may be used in future 8051 products to invoke new features.

Philips Semiconductors

80C51 Family — 80C51 family programmer's guide and instruction set

INTERRUPTS:

To use any of the interrupts in the 80C51 Family, the following three steps must be taken.

1. **Set the EA (enable all) bit in the IE register to 1.**
2. **Set the corresponding individual interrupt enable bit in the IE register to 1.**
3. **Begin the interrupt service routine at the corresponding Vector Address of that interrupt. See Table below.**

INTERRUPT SOURCE	VECTOR ADDRESS
IE0	0003H
TF0	000BH
IE1	0013H
TF1	001BH
RI & TI	0023H

In addition, for external interrupts, pins INT0 and INT1 (P3.2 and P3.3) must be set to 1, and depending on whether the interrupt is to be level or transition activated, bits IT0 or IT1 in the TCON register may need to be set to 1.

ITx = 0 level activated

ITx = 1 transition activated

IE: INTERRUPT ENABLE REGISTER. BIT ADDRESSABLE.

If the bit is 0, the corresponding interrupt is disabled. If the bit is 1, the corresponding interrupt is enabled.

EA	–	–	ES	ET1	EX1	ET0	EX0

EA IE.7 Disables all interrupts. If EA = 0, no interrupt will be acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
— IE.6 Not implemented, reserved for future use.*
— IE.5 Not implemented, reserved for future use.*
ES IE.4 Enable or disable the serial port interrupt.
ET1 IE.3 Enable or disable the Timer 1 overflow interrupt.
EX1 IE.2 Enable or disable External Interrupt 1.
ET0 IE.1 Enable or disable the Timer 0 overflow interrupt.
EX0 IE.0 Enable or disable External Interrupt 0.

* User software should not write 1s to reserved bits. These bits may be used in future 80C51 products to invoke new features.

ASSIGNING HIGHER PRIORITY TO ONE OR MORE INTERRUPTS:

In order to assign higher priority to an interrupt the corresponding bit in the IP register must be set to 1.

Remember that while an interrupt service is in progress, it cannot be interrupted by a lower or same level interrupt.

PRIORITY WITHIN LEVEL:

Priority within level is only to resolve simultaneous requests of the same priority level.

From high to low, interrupt sources are listed below:

IE0
TF0
IE1
TF1
RI or TI

IP: INTERRUPT PRIORITY REGISTER. BIT ADDRESSABLE.

If the bit is 0, the corresponding interrupt has a lower priority and if the bit is 1 the corresponding interrupt has a higher priority.

–	–	–	PS	PT1	PX1	PT0	PX0

–	IP.7	Not implemented, reserved for future use.*
–	IP.6	Not implemented, reserved for future use.*
–	IP.5	Not implemented, reserved for future use.*
PS	IP.4	Defines the Serial Port interrupt priority level.
PT1	IP.3	Defines the Timer 1 interrupt priority level.
PX1	IP.2	Defines External Interrupt 1 priority level.
PT0	IP.1	Defines the Timer 0 interrupt priority level.
PX0	IP.0	Defines the External Interrupt 0 priority level.

* User software should not write 1s to reserved bits. These bits may be used in future 80C51 products to invoke new features.

TCON: TIMER/COUNTER CONTROL REGISTER. BIT ADDRESSABLE.

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1	TCON.7	Timer 1 overflow flag. Set by hardware when the Timer/Counter 1 overflows. Cleared by hardware as processor vectors to the interrupt service routine.
TR1	TCON.6	Timer 1 run control bit. Set/cleared by software to turn Timer/Counter 1 ON/OFF.
TF0	TCON.5	Timer 0 overflow flag. Set by hardware when the Timer/Counter 0 overflows. Cleared by hardware as processor vectors to the service routine.
TR0	TCON.4	Timer 0 run control bit. Set/cleared by software to turn Timer/Counter 0 ON/OFF.
IE1	TCON.3	External Interrupt 1 edge flag. Set by hardware when External Interrupt edge is detected. Cleared by hardware when interrupt is processed.
IT1	TCON.2	Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt.
IE0	TCON.1	External Interrupt 0 edge flag. Set by hardware when External Interrupt edge detected. Cleared by hardware when interrupt is processed.
IT0	TCON.0	Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt.

TMOD: TIMER/COUNTER MODE CONTROL REGISTER. NOT BIT ADDRESSABLE.

GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
Timer 1				Timer 0			

GATE	When TRx (in TCON) is set and GATE = 1, TIMER/COUNTERx will run only while INTx pin is high (hardware control). When GATE = 0, TIMER/COUNTERx will run only while TRx = 1 (software control).
C/$\overline{T}$	Timer or Counter selector. Cleared for Timer operation (input from internal system clock). Set for Counter operation (input from Tx input pin).
M1	Mode selector bit. (NOTE 1)
M0	Mode selector bit. (NOTE 1)

NOTE 1:

M1	M0	Operating Mode	
0	0	0	13-bit Timer (8048 compatible)
0	1	1	16-bit Timer/Counter
1	0	2	8-bit Auto-Reload Timer/Counter
1	1	3	(Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standart Timer 0 control bits. TH0 is an8-bit Timer and is controlled by Timer 1 control bits.
1	1	3	(Timer 1) Timer/Counter 1 stopped.

TIMER SET-UP

Tables 2 through 5 give some values for TMOD which can be used to set up Timer 0 in different modes.

It is assumed that only one timer is being used at a time. If it is desired to run Timers 0 and 1 simultaneously, in any mode, the value in TMOD for Timer 0 must be ORed with the value shown for Timer 1 (Tables 5 and 6).

For example, if it is desired to run Timer 0 in mode 1 GATE (external control), and Timer 1 in mode 2 COUNTER, then the value that must be loaded into TMOD is 69H (09H from Table 2 ORed with 60H from Table 5).

Moreover, it is assumed that the user, at this point, is not ready to turn the timers on and will do that at a different point in the program by setting bit TRx (in TCON) to 1.

TIMER/COUNTER 0

Table 2. As a Timer:

MODE	TIMER 0 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	00H	08H
1	16-bit Timer	01H	09H
2	8-bit Auto-Reload	02H	0AH
3	Two 8-bit Timers	03H	0BH

Table 3. As a Counter:

MODE	COUNTER 0 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	04H	0CH
1	16-bit Timer	05H	0DH
2	8-bit Auto-Reload	06H	0EH
3	One 8-bit Counter	07H	0FH

NOTES:
1. The timer is turned ON/OFF by setting/clearing bit TR0 in the software.
2. The Timer is turned ON/OFF by the 1-to-0 transition on $\overline{INT0}$ (P3.2) when TR0 = 1 (hardware control).

TIMER/COUNTER 1

Table 4. As a Timer:

MODE	TIMER 1 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	00H	80H
1	16-bit Timer	10H	90H
2	8-bit Auto-Reload	20H	A0H
3	Does not run	30H	B0H

Table 5. As a Counter:

MODE	COUNTER 1 FUNCTION	TMOD	
		INTERNAL CONTROL (NOTE 1)	EXTERNAL CONTROL (NOTE 2)
0	13-bit Timer	40H	C0H
1	16-bit Timer	50H	D0H
2	8-bit Auto-Reload	60H	E0H
3	Not available	–	–

NOTES:
1. The timer is turned ON/OFF by setting/clearing bit TR1 in the software.
2. The Timer is turned ON/OFF by the 1-to-0 transition on $\overline{INT1}$ (P3.2) when TR1 = 1 (hardware control).

SCON: SERIAL PORT CONTROL REGISTER. BIT ADDRESSABLE.

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

SM0	SCON.7	Serial Port mode specifier. (NOTE 1)
SM1	SCON.6	Serial Port mode specifier. (NOTE 1)
SM2	SCON.5	Enables the multiprocessor communication feature in modes 2 & 3. In mode 2 or 3, if SM2 is set to 1 then RI will not be activated if the received 9th data bit (RB8) is 0. In mode 1, if SM2 = 1 then RI will not be activated if a valid stop bit was not received. In mode 0, SM2 should be 0. (See Table 6.)
REN	SCON.4	Set/Cleared by software to Enable/Disable reception.
TB8	SCON.3	The 9th bit that will be transmitted in modes 2 & 3. Set/Cleared by software.
RB8	SCON.2	In modes 2 & 3, is the 9th data bit that was received. In mode 1, if SM2 = 0, RB8 is the stop bit that was received. In mode 0, RB8 is not used.
TI	SCON.1	Transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the beginning of the stop bit in the other modes. Must be cleared by software.
RI	SCON.0	Receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or halfway through the stop bit time in the other modes (except see SM2). Must be cleared by software.

NOTE 1:

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Shift Register	$F_{OSC.}/12$
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	$F_{OSC.}/64$ or $F_{OSC.}/32$
1	1	3	9-bit UART	Variable

SERIAL PORT SET-UP:

Table 6.

MODE	SCON	SM2 VARIATION
0 1 2 3	10H 50H 90H D0H	Single Processor Environment (SM2 = 0)
0 1 2 3	NA 70H B0H F0H	Multiprocessor Environment (SM2 = 1)

GENERATING BAUD RATES

Serial Port in Mode 0:

Mode 0 has a fixed baud rate which is 1/12 of the oscillator frequency. To run the serial port in this mode none of the Timer/Counters need to be set up. Only the SCON register needs to be defined.

$$\text{Baud Rate} \quad \frac{\text{Osc Freq}}{12}$$

Serial Port in Mode 1:

Mode 1 has a variable baud rate. The baud rate is generated by Timer 1.

USING TIMER/COUNTER 1 TO GENERATE BAUD RATES:

For this purpose, Timer 1 is used in mode 2 (Auto-Reload). Refer to Timer Setup section of this chapter.

$$\text{Baud Rate} \quad \frac{K}{32} \quad \frac{\text{Osc Freq}}{12 \quad [256 \quad (TH1)]}$$

If SMOD = 0, then K = 1.
If SMOD = 1, then K = 2 (SMOD is in the PCON register).

Most of the time the user knows the baud rate and needs to know the reload value for TH1.

$$TH1 \quad 256 \quad \frac{K \quad \text{Osc Freq}}{384 \quad \text{baud rate}}$$

TH1 must be an integer value. Rounding off TH1 to the nearest integer may not produce the desired baud rate. In this case, the user may have to choose another crystal frequency.

Since the PCON register is not bit addressable, one way to set the bit is logical ORing the PCON register (i.e., ORL PCON,#80H). The address of PCON is 87H.

SERIAL PORT IN MODE 2:

The baud rate is fixed in this mode and is 1/32 or 1/64 of the oscillator frequency, depending on the value of the SMOD bit in the PCON register.

In this mode none of the Timers are used and the clock comes from the internal phase 2 clock.

SMOD = 1, Baud Rate = 1/32 Osc Freq.

SMOD = 0, Baud Rate = 1/64 Osc Freq.

To set the SMOD bit: ORL PCON,#80H. The address of PCON is 87H.

SERIAL PORT IN MODE 3:

The baud rate in mode 3 is variable and sets up exactly the same as in mode 1.

80C51 FAMILY INSTRUCTION SET

Table 7. 80C51 Instruction Set Summary

Interrupt Response Time: Refer to Hardware Description Chapter.

Instructions that Affect Flag Settings[(1)]

Instruction	C	OV	AC	Instruction	C	OV	AC
ADD	X	X	X	CLR C	0		
ADDC	X	X	X	CPL C	X		
SUBB	X	X	X	ANL C,bit	X		
MUL	0	X		ANL C,/bit	X		
DIV	0	X		ORL C,bit	X		
DA	X			ORL C,/bit	X		
RRC	X			MOV C,bit	X		
RLC	X			CJNE	X		
SETB C	1						

[(1)]Note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

Notes on instruction set and addressing modes:

Rn	Register R7-R0 of the currently selected Register Bank.
direct	8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)].
@Ri	8-bit internal data RAM location (0-255) addressed indirectly through register R1 or R0.
#data	8-bit constant included in the instruction.
#data 16	16-bit constant included in the instruction
addr 16	16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64k-byte Program Memory address space.
addr 11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2k-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is –128 to +127 bytes relative to first byte of the following instruction.
bit	Direct Addressed bit in Internal Data RAM or Special Function Register.

MNEMONIC		DESCRIPTION	BYTE	OSCILLATOR PERIOD
ARITHMETIC OPERATIONS				
ADD	A,Rn	Add register to Accumulator	1	12
ADD	A,direct	Add direct byte to Accumulator	2	12
ADD	A,@Ri	Add indirect RAM to Accumulator	1	12
ADD	A,#data	Add immediate data to Accumulator	2	12
ADDC	A,Rn	Add register to Accumulator with carry	1	12
ADDC	A,direct	Add direct byte to Accumulator with carry	2	12
ADDC	A,@Ri	Add indirect RAM to Accumulator with carry	1	12
ADDC	A,#data	Add immediate data to A_{CC} with carry	2	12
SUBB	A,Rn	Subtract Register from A_{CC} with borrow	1	12
SUBB	A,direct	Subtract direct byte from A_{CC} with borrow	2	12
SUBB	A,@Ri	Subtract indirect RAM from A_{CC} with borrow	1	12
SUBB	A,#data	Subtract immediate data from A_{CC} with borrow	2	12
INC	A	Increment Accumulator	1	12
INC	Rn	Increment register	1	12

Philips Semiconductors
80C51 Family — 80C51 family programmer's guide and instruction set

Table 7. 80C51 Instruction Set Summary (Continued)

MNEMONIC		DESCRIPTION	BYTE	OSCILLATOR PERIOD
ARITHMETIC OPERATIONS (Continued)				
INC	direct	Increment direct byte	2	12
INC	@Ri	Increment indirect RAM	1	12
DEC	A	Decrement Accumulator	1	12
DEC	Rn	Decrement Register	1	12
DEC	direct	Decrement direct byte	2	12
DEC	@Ri	Decrement indirect RAM	1	12
INC	DPTR	Increment Data Pointer	1	24
MUL	AB	Multiply A and B	1	48
DIV	AB	Divide A by B	1	48
DA	A	Decimal Adjust Accumulator	1	12
LOGICAL OPERATIONS				
ANL	A,Rn	AND Register to Accumulator	1	12
ANL	A,direct	AND direct byte to Accumulator	2	12
ANL	A,@Ri	AND indirect RAM to Accumulator	1	12
ANL	A,#data	AND immediate data to Accumulator	2	12
ANL	direct,A	AND Accumulator to direct byte	2	12
ANL	direct,#data	AND immediate data to direct byte	3	24
ORL	A,Rn	OR register to Accumulator	1	12
ORL	A,direct	OR direct byte to Accumulator	2	12
ORL	A,@Ri	OR indirect RAM to Accumulator	1	12
ORL	A,#data	OR immediate data to Accumulator	2	12
ORL	direct,A	OR Accumulator to direct byte	2	12
ORL	direct,#data	OR immediate data to direct byte	3	24
XRL	A,Rn	Exclusive-OR register to Accumulator	1	12
XRL	A,direct	Exclusive-OR direct byte to Accumulator	2	12
XRL	A,@Ri	Exclusive-OR indirect RAM to Accumulator	1	12
XRL	A,#data	Exclusive-OR immediate data to Accumulator	2	12
XRL	direct,A	Exclusive-OR Accumulator to direct byte	2	12
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	24
CLR	A	Clear Accumulator	1	12
CPL	A	Complement Accumulator	1	12
RL	A	Rotate Accumulator left	1	12
RLC	A	Rotate Accumulator left through the carry	1	12
RR	A	Rotate Accumulator right	1	12
RRC	A	Rotate Accumulator right through the carry	1	12
SWAP	A	Swap nibbles within the Accumulator	1	12
DATA TRANSFER				
MOV	A,Rn	Move register to Accumulator	1	12
MOV	A,direct	Move direct byte to Accumulator	2	12
MOV	A,@Ri	Move indirect RAM to Accumulator	1	12

Table 7. 80C51 Instruction Set Summary (Continued)

MNEMONIC		DESCRIPTION	BYTE	OSCILLATOR PERIOD
DATA TRANSFER (Continued)				
MOV	A,#data	Move immediate data to Accumulator	2	12
MOV	Rn,A	Move Accumulator to register	1	12
MOV	Rn,direct	Move direct byte to register	2	24
MOV	RN,#data	Move immediate data to register	2	12
MOV	direct,A	Move Accumulator to direct byte	2	12
MOV	direct,Rn	Move register to direct byte	2	24
MOV	direct,direct	Move direct byte to direct	3	24
MOV	direct,@Ri	Move indirect RAM to direct byte	2	24
MOV	direct,#data	Move immediate data to direct byte	3	24
MOV	@Ri,A	Move Accumulator to indirect RAM	1	12
MOV	@Ri,direct	Move direct byte to indirect RAM	2	24
MOV	@Ri,#data	Move immediate data to indirect RAM	2	12
MOV	DPTR,#data16	Load Data Pointer with a 16-bit constant	3	24
MOVC	A,@A+DPTR	Move Code byte relative to DPTR to A_{CC}	1	24
MOVC	A,@A+PC	Move Code byte relative to PC to A_{CC}	1	24
MOVX	A,@Ri	Move external RAM (8-bit addr) to A_{CC}	1	24
MOVX	A,@DPTR	Move external RAM (16-bit addr) to A_{CC}	1	24
MOVX	A,@Ri,A	Move A_{CC} to external RAM (8-bit addr)	1	24
MOVX	@DPTR,A	Move A_{CC} to external RAM (16-bit addr)	1	24
PUSH	direct	Push direct byte onto stack	2	24
POP	direct	Pop direct byte from stack	2	24
XCH	A,Rn	Exchange register with Accumulator	1	12
XCH	A,direct	Exchange direct byte with Accumulator	2	12
XCH	A,@Ri	Exchange indirect RAM with Accumulator	1	12
XCHD	A,@Ri	Exchange low-order digit indirect RAM with A_{CC}	1	12
BOOLEAN VARIABLE MANIPULATION				
CLR	C	Clear carry	1	12
CLR	bit	Clear direct bit	2	12
SETB	C	Set carry	1	12
SETB	bit	Set direct bit	2	12
CPL	C	Complement carry	1	12
CPL	bit	Complement direct bit	2	12
ANL	C,bit	AND direct bit to carry	2	24
ANL	C,/bit	AND complement of direct bit to carry	2	24
ORL	C,bit	OR direct bit to carry	2	24
ORL	C,/bit	OR complement of direct bit to carry	2	24
MOV	C,bit	Move direct bit to carry	2	12
MOV	bit,C	Move carry to direct bit	2	24
JC	rel	Jump if carry is set	2	24
JNC	rel	Jump if carry not set	2	24

Philips Semiconductors

80C51 Family — 80C51 family programmer's guide and instruction set

Table 7. 80C51 Instruction Set Summary (Continued)

MNEMONIC		DESCRIPTION	BYTE	OSCILLATOR PERIOD
BOOLEAN VARIABLE MANIPULATION (Continued)				
JB	rel	Jump if direct bit is set	3	24
JNB	rel	Jump if direct bit is not set	3	24
JBC	bit,rel	Jump if direct bit is set and clear bit	3	24
PROGRAM BRANCHING				
ACALL	addr11	Absolute subroutine call	2	24
LCALL	addr16	Long subroutine call	3	24
RET		Return from subroutine	1	24
RETI		Return from interrupt	1	24
AJMP	addr11	Absolute jump	2	24
LJMP	addr16	Long jump	3	24
SJMP	rel	Short jump (relative addr)	2	24
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	24
JZ	rel	Jump if Accumulator is zero	2	24
JNZ	rel	Jump if Accumulator is not zero	2	24
CJNE	A,direct,rel	Compare direct byte to A_{CC} and jump if not equal	3	24
CJNE	A,#data,rel	Compare immediate to A_{CC} and jump if not equal	3	24
CJNE	RN,#data,rel	Compare immediate to register and jump if not equal	3	24
CJNE	@Ri,#data,rel	Compare immediate to indirect and jump if not equal	3	24
DJNZ	Rn,rel	Decrement register and jump if not zero	2	24
DJNZ	direct,rel	Decrement direct byte and jump if not zero	3	24
NOP		No operation	1	12

APPENDICE C: INDICE ANALITICO

M

P

Q

R

S